

自由飞行空间机器人 运动控制及仿真

洪炳镕 柳长安 刘宏 著

国防工业出版社

· 北京 ·

虑这种影响,而仍然采用地面固定基座机器人的控制技术,空间机器人将不但无法完成预定的捕捉目标或更换零部件等操作任务,而且由于姿态的变化将会破坏卫星的正常工作。为克服这个问题,必须找到能够考虑这种相互作用的运动学建模方法和控制算法。另外,云云在完成任
务时必须对某一目标进行操作,因此必须首先飞行到操作可能的会合点位置,然后再用机械臂进行操作,因此对云云来说,不但要研究卫星本体运动的路径规划,而且还要研究机械臂末端效应器的路径规划。

本书是在分析和总结国内外近年来云云相关研究的基础上,综合数名博士刘宏、朱铁一、吴威、何光彩、李华忠、王鸿鹏及柳长安等人的博士论文研究成果撰写而成的。本书由柳长安和刘宏两位博士负责各个章节的撰写,再由本人进行整体的编辑和修改。在专著撰写过程中得到了中国航天局张履谦院士的热情支持,在此表示衷心感谢。

本书内容虽然针对自由飞行空间机器人,研究其运动学模型、姿态控制、飞行和操作所需的最佳路径以及多机器人的协调等,但其研究方法也适用于其它类型的空间机器人,尤其是本专著中各个章节提出的仿真学研究方法也适用于其它空间机器人的研究。

自由飞行空间机器人技术是面向云世纪的一门高新技术,正处在飞速发展之中,如果本书能对研究和开发空间机器人的科学家和工程技术人员起到参考作用我们将感到十分荣幸。

洪炳镕
于哈尔滨工业大学
云年 怨月 员日

目 录

第一章 绪论	员
1.1 空间机器人的概念及分类	员
1.2 云砸的主要用途及研究意义	缘
1.3 云砸的研究现状	远
1.4 基本理论的研究	远
1.5 地面实验平台的研制	愿
1.6 空间试验系统的开发	怨
1.7 本书概要	圆
第二章 云砸运动学模型及运动特性	圆
2.1 云砸的运动学模型	圆
2.2 云砸的模型及参数定义	圆
2.3 云砸运动学方程	猿
2.4 云砸通用运动学模型	圆
2.5 云砸运动学方程	圆
2.6 云砸运动学分析	猿
2.7 云砸的运动特性	圆
2.8 姿态干扰特性	圆
2.9 冗余特性	猿
2.10 非完整性	猿
2.11 云砸的工作空间及其划分	圆
2.12 云砸工作空间	圆
2.13 云砸工作空间划分	圆
2.14 连杆质量对工作空间的影响	圆
2.15 多臂云砸运动学模型	圆
2.16 云砸运动学模型	圆
2.17 云砸运动学方程	猿
2.18 小结	猿
第三章 基于神经网络的空间机器人参数辨识	猿
3.1 前馈神经网络的逼近能力和学习算法	猿
3.2 多层前馈神经网络的逼近能力	猿
3.3 多层前馈神经网络的学习算法	猿

摇摇猿猿猿猿猿猿猿猿非线性同伦综合学习算法.....	猿猿
摇摇猿猿猿猿猿猿猿猿同伦 月孕学习算法	猿猿
摇摇猿猿猿猿猿猿猿猿非线性同伦 月孕学习算法	猿猿
摇摇猿猿猿猿猿猿猿猿非线性同伦综合学习算法.....	猿猿
摇摇猿猿猿猿猿猿猿猿各种算法的性能比较.....	猿猿
摇摇猿猿猿猿猿猿猿猿参数辨识	猿猿
摇摇猿猿猿猿猿猿猿猿基于神经网络的参数辨识算法.....	猿猿
摇摇猿猿猿猿猿猿猿猿计算机仿真.....	猿猿
摇摇猿猿猿猿猿猿猿猿小结.....	猿猿
第四章 摇摇猿猿猿猿猿猿捕捉目标运动规划	猿猿
摇摇猿猿猿猿猿猿猿猿运动规划	猿猿
摇摇猿猿猿猿猿猿猿猿控制算法	猿猿
摇摇猿猿猿猿猿猿猿猿分解运动速度控制.....	猿猿
摇摇猿猿猿猿猿猿猿猿分解运动加速度控制.....	猿猿
摇摇猿猿猿猿猿猿猿猿机械臂典型运动分析.....	猿猿
摇摇猿猿猿猿猿猿猿猿捕捉静态目标路径规划	猿猿
摇摇猿猿猿猿猿猿猿猿捕捉静态目标策略.....	猿猿
摇摇猿猿猿猿猿猿猿猿捕捉静态目标路径规划算法.....	猿猿
摇摇猿猿猿猿猿猿猿猿计算机仿真.....	猿猿
摇摇猿猿猿猿猿猿猿猿捕捉动态目标路径规划	猿猿
摇摇猿猿猿猿猿猿猿猿捕捉运动目标策略.....	猿猿
摇摇猿猿猿猿猿猿猿猿捕捉运动目标路径规划算法.....	猿猿
摇摇猿猿猿猿猿猿猿猿计算仿真.....	猿猿
摇摇猿猿猿猿猿猿猿猿避免碰撞路径规划	猿猿
摇摇猿猿猿猿猿猿猿猿双向双启发函数.....	猿猿
摇摇猿猿猿猿猿猿猿猿避免碰撞路径规划算法.....	猿猿
摇摇猿猿猿猿猿猿猿猿计算机仿真.....	猿猿
摇摇猿猿猿猿猿猿猿猿小结.....	猿猿
第五章 摇摇基于受限最小干扰图的姿态控制	猿猿
摇摇猿猿猿猿猿猿猿猿的姿态控制方法	猿猿
摇摇猿猿猿猿猿猿猿猿干扰图与增强干扰图.....	猿猿
摇摇猿猿猿猿猿猿猿猿干扰图.....	猿猿
摇摇猿猿猿猿猿猿猿猿增强干扰图.....	猿猿
摇摇猿猿猿猿猿猿猿猿姿态干扰计算及受限最小干扰图.....	猿猿
摇摇猿猿猿猿猿猿猿猿姿态干扰计算.....	猿猿
摇摇猿猿猿猿猿猿猿猿受限最小干扰图.....	猿猿
摇摇猿猿猿猿猿猿猿猿砸云云图与 耘云云图的比较	猿猿
摇摇猿猿猿猿猿猿猿猿基于 砸云云云云的姿态控制算法	猿猿
摇摇猿猿猿猿猿猿猿猿零姿态干扰的计算.....	猿猿

摇摇摇缘缘缘基于 砸云砸的姿态控制算法	远苑
摇摇摇缘缘缘计算机仿真	远怨
摇摇缘缘缘小结	远园
第六章摇摇基于姿态稳定的冗余 云云砸运动规划	苑园
摇摇摇摇基于姿态稳定的冗余多臂 云云砸运动分析	苑园
摇摇摇摇缘缘缘云云砸冗余性分析	苑园
摇摇摇摇缘缘缘姿态受限广义雅可比矩阵	苑园
摇摇摇摇缘缘缘分解运动速度控制	苑源
摇摇摇摇缘缘缘动力学奇异点与回避	苑缘
摇摇摇摇缘缘缘基于 粤云砸的 云云砸运动规划	苑缘
摇摇摇摇缘缘缘自由浮游状态下捕捉目标运动规划	苑缘
摇摇摇摇缘缘缘自由飞行状态下捕捉目标运动规划	苑远
摇摇摇摇缘缘缘计算机仿真	苑苑
摇摇缘缘缘小结	愿
第七章摇摇云云砸姿态控制综合算法	愿
摇摇苑苑苑卫星的姿态控制	愿
摇摇苑苑苑基于姿态干扰预测的姿态控制算法	愿
摇摇苑苑苑苑苑苑苑基本思想	愿
摇摇苑苑苑苑苑苑苑理论分析	愿
摇摇苑苑苑苑苑苑苑计算机仿真	愿
摇摇苑苑苑云云砸复合姿态控制算法	愿
摇摇苑苑苑苑苑苑苑复合姿态控制算法	愿
摇摇苑苑苑苑苑苑苑捕捉目标后的空间作业运动规划	愿
摇摇苑苑苑缘缘缘小结	愿
第八章摇摇基于关节力矩的 云云砸运动控制算法	愿
摇摇愿愿愿关节驱动力矩递推算法	愿
摇摇摇摇愿愿愿运动模型	愿
摇摇摇摇愿愿愿云云砸的速度分析	愿
摇摇摇摇愿愿愿云云砸的加速度分析	愿
摇摇摇摇愿愿愿云云砸的关节驱动力矩计算公式	愿
摇摇愿愿愿基于关节力矩的捕捉目标控制算法	愿
摇摇愿愿愿苑苑苑苑苑苑苑控制策略	愿
摇摇愿愿愿苑苑苑苑苑苑苑云云砸捕捉目标控制算法	愿
摇摇愿愿愿计算机仿真	愿
摇摇愿愿缘缘缘小结	愿
第九章摇摇云云砸模糊运动控制算法	愿
摇摇愿愿愿模糊逻辑和分级模糊控制器	愿
摇摇愿愿愿基于模糊规则的机器人运动控制算法	愿
摇摇愿愿愿苑苑苑苑苑苑苑二自由度机器人的基于模糊规则的控制设计	愿

三连杆 云端的基于模糊规则的控制器的设计	页
基于模糊逻辑的 云端路径规划算法	页
计算机仿真	页
小结	页
第十章 云端的飞行轨迹优化	页
云端的飞行轨迹优化模型	页
单 云端接近静止目标飞行轨迹优化	页
单 云端截击运动目标飞行轨迹优化	页
多 云端飞行轨迹优化	页
参数优化方法	页
计算机仿真	页
单 云端接近静止目标飞行轨迹优化仿真	页
单 云端截击运动目标飞行轨迹优化仿真	页
多 云端飞行轨迹优化仿真	页
小结	页
第十一章 多个 云端协调操作控制算法	页
多 云端运动学模型	页
多 云端运动学模型	页
多 云端运动学方程	页
多 云端动力学模型	页
多 云端关节力递推式	页
多 云端动力学方程	页
多 云端协调控制算法	页
多 云端协调控制方案	页
多 云端稳定性分析	页
多 云端计算机仿真	页
多 云端小结	页
第十二章 云端地面实验平台	页
地面实验平台系统组成	页
机器人模型	页
主控板	页
控制板总体方案设计	页
直流电机的模糊控制算法	页
视觉系统	页
云端地面实验平台全局视觉方案	页
视觉系统软件的工作原理	页
操作台	页
操作台显示信息	页
操作台命令	页

摇摇员规划系统.....	员缘
摇摇摇摇员捕捉策略.....	员缘
摇摇摇摇员基于广义雅可比矩阵捕捉目标的运动规划.....	员远
摇摇摇摇员基于姿态受限广义雅可比矩阵捕捉目标的运动规划.....	员远
摇摇员无线通信系统.....	员苑
摇摇员网络系统.....	员愿
摇摇摇摇员网络系统构成.....	员愿
摇摇摇摇员服务器端的框架.....	员愿
摇摇摇摇员控制台结构.....	员愿
摇摇员气浮系统.....	员园
摇摇员仿真实验.....	员园
摇摇摇摇员捕捉目标.....	员园
摇摇摇摇员操作目标.....	员员
摇摇摇摇员对接目标.....	员员
摇摇员小结.....	员圆
参考文献.....	员猿

第一章 绪 论

近年来,随着科学技术的飞速发展,经过科学家们坚持不懈的努力,人类遨游宇宙太空的梦想正逐步得以实现,如发射卫星,漫步火星、月球,建立地外空间站等。宇宙已经成为人类另一个生存和活动的空间。宇宙空间为人类提供了特殊的生产基地和科学实验场所。研究和开发太空资源已经成为人类文明发展的一个重要标志。然而恶劣的空间环境,如放射线、高温差、超真空等给人类在太空的生存和活动带来了巨大的威胁。要使人类在太空停留,需要庞大而复杂的环境控制系统、生命保障系统、物质补给系统、救生系统等,这些系统耗资巨大。

在未来的空间活动中,如交会对接、卫星捕获及大量的空间加工、生产、装配、检测、维修和空间科学实验等任务,仅靠航天员是完成不了的。世界各国空间技术研究人員愈来愈清楚地认识到,采用空间机器人协助航天员完成太空作业、太空实验和太空考察是当今行之有效的最佳选择。

空间机器人学作为一门新兴的边缘学科,涉及到计算机、人工智能、自动控制、无线电通信、传感器融合、人工生命、机械学和力学等多个研究领域。目前美国、日本、加拿大、德国和法国等都在加紧进行空间机器人的研制和实验工作。我国空间机器人的研究起步比较晚,因此必须跟踪世界先进水平,加速发展空间机器人的研究工作。

1.1 空间机器人的概念及分类

空间机器人(Space Robots, SR)是在太空环境中运用的特种机器人,主要应用于在轨装配与服务、科学实验载荷照料以及星球表面探测等方面。空间机器人技术已成为人类探索、开发和利用太空资源的重要手段,也是当今衡量一个国家高科技发展水平的重要标志。空间机器人将在扩大人类空间活动能力、提高安全性、提高生产效率和经济效益等方面发挥巨大的作用。根据不同的标准和原则,空间机器人有多种分类方法。

1. 按用途划分

1) 行星探测机器人

发展行星探测机器人的目的是满足行星表面探测任务的需要。在行星探测中,机器人将被用来探测可能的着陆地点、感兴趣的科学考查地域、放置科学仪器、收集用来分析并可能被送回地球的物质样品。行星探测需要机器人具有高度的自主性,包括区域自主导航、确定有潜在科学考查价值的地域、管理自身资源并制定活动计划,所有这些工作仅需少量的地面指令。其研究的重点是移动系统、微型机械装置、任务规划和自主导航。火星探路者(Mars Pathfinder)就是该种机器人,它是1996年12月4日从肯尼迪空间中心发射

的,经过7个月的长途跋涉,于1997年7月4日登上火星。登陆器中装载有漫游机器人,取名为索杰纳(Sojourner),如图1-1所示。火星探路者登上火星后,索杰纳离开登陆器在火星表面漫游,移动范围只有几十米。该漫游机器人主要用做技术实验,验证微型漫游机器人在火星表面的运动特性,以确保将来的火星漫游机器人具有更有效的导航和运动性能。

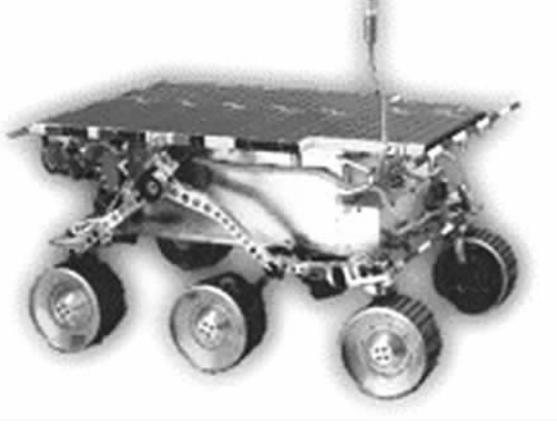


图 1-1 火星漫游机器人索杰纳

2) 舱内服务机器人

舱内服务机器人主要用于在舱内进行科学实验的维护和照料。它可以代替航天员的一部分工作,以减轻航天员的负担。舱内服务机器人需要质量小和体积小,具有冗余度和灵活性,它主要作为空间站舱内使用的机器人,如图1-2所示。发展舱内服务机器人的目的在于降低科学试验载荷对航天员的依赖性,甚至航天员不在现场的情况下也能对科学载荷进行照管。舱内服务机器人最典型的应用是国际空间站舱内活动机器人和美国天空实验中的实验看管机器人。



图 1-2 空间站舱内服务机器人

3) 附着式服务机器人

附着式服务机器人的特点是带着一个附着系统,可以连接在航天飞机或空间站等载体上,这类机器人主要用来提供在轨服务,包括小型卫星维修、空间平台外部有效载荷服务。比较典型的有美国航天飞机遥控机器人系统,如图 1-3 所示。该系统的机械臂是加拿大制造的。机械臂全长 15m,具有 6 个自由度,各关节由无刷直流电机及齿轮减速驱动,臂上装有视觉相机,整个系统安装在航天飞机货舱口上,控制系统位于驾驶舱内,主要包括:计算机控制系统、监视 / 显示器、控制板面及两个三自由度手控器。该系统具有主从控制及局部自主控制两种模式。自 1981 年起已在空间多次成功地进行了轨道飞行器的组装、维修、回收和释放等操作,如 1984 年利用该系统帮助修复了玛克希姆太阳观测仪,1997 年利用该系统成功地修复了哈勃望远镜。同时,它也将是国际空间站初试组装阶段的主要装配工具。

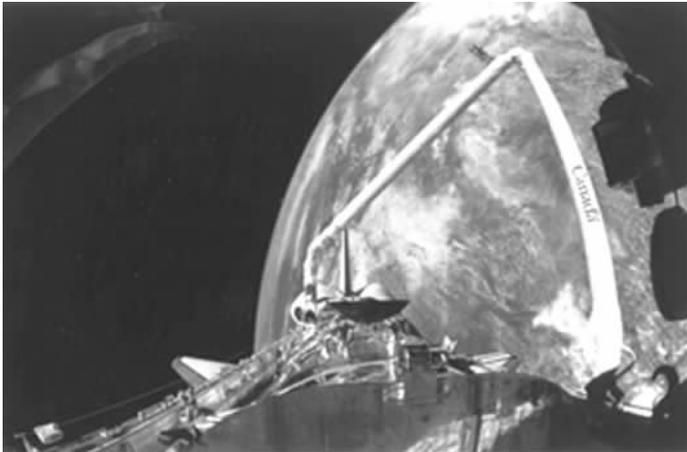


图 1-3 航天飞机遥控机器人系统

4) 自由飞行机器人

自由飞行空间机器人(Free-Flying Space Robot,FFSR) 主要应用于卫星的在轨服务,这个任务目前是由穿充压宇航服的宇航员来完成的。所需的技术包括:具有大延时的遥操作和遥现技术、虚拟现实技术、先进的显示技术、机器人故障检测、人—机器人协调控制、自由飞行巡视和检测技术、地面控制技术。美国马里兰大学从 1992 年起开始了漫游者空间机器人(Ranger)的设计与研究,1996 年 10 月完成了遥控机器人飞行实验,Ranger 的原型系统如图 1-4 所示。它由 4 个机械手组成:两个 8 自由度手臂用于侧向灵活操作,一个 7 自由度机械臂用于当地的固定,一个 6 自由度手臂用于为一对立体视觉摄像机定位,从而为远程操作提供视觉反馈。另外还有一对摄像机安装在 Ranger 的中心,为机动飞行提供稳定的视觉参考和作为自主对接的视觉系统。

2. 按控制方式划分

1) 主从式遥控机械手

主从式遥控机械手(Master-Slave Teleoperator)由主手和从手组成。操作人员通过主手控制从手的动作。美国早期的航天器上的机器人都属于这种类型,这种机械手具有很多缺点。首先,操作人员是整个控制回路的组成部分,操作人员的劳动强度很大,短时间的操

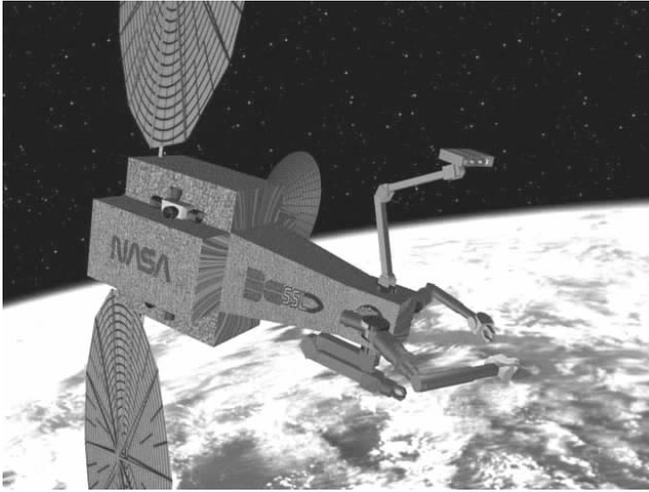


图 1-4 漫游者空间机器人

作就可以使操作人员感到很疲惫。其次,进行遥控操作时,控制信号的传输与反馈存在延时,延时大到一定的程度时系统就不稳定。因此,早期航天器上的遥控机械手都采用发信→等待→发信的开环控制方式,以解决系统的稳定性问题。这种形式的遥控机械手已经逐渐被遥控机器人所代替。

2) 遥控机器人

遥控机器人(Telerobot)已经不是一种简单的操作手,这种机器人是将遥控和一定程度的自主技术相结合的机器人系统。该系统有两个回路:本地控制回路和远地控制回路,在两个回路之间进行通信。远地回路由具有智能传感器的机器人和远地计算机构成,远地计算机根据操作人员发出的指令控制机器人工作。这个回路是低智能和高响应速率(即高宽带)的回路。操作人员工作在本地回路中,借助遥现装置发来的各种信息,利用操作人员的高智能监控远地回路中机器人的工作。这个回路是高智能和低响应速率(即低宽带)的回路。这种空间机器人的智能还不高,适应环境的能力还不强。目前,在航天飞机上和早期空间站上所使用的机器人多属于这种类型。遥控机器人可以工作在舱内,也可以工作在舱外,还可以安装在空间自由飞行器上到远离空间站的地方去执行任务。

3) 自主式机器人

自主式机器人(Autonomous Robot)是一种应用智能控制的机器人,无需人类干预就能自主完成其预定任务。这种机器人具有类似人的视觉、触觉、力觉、接近觉和听觉等感知功能,能感知外界环境的变化并自动适应外界环境。自主式机器人具有规划、编程和诊断功能,具有修理其他机器人和自修理的功能。自主式机器人尚处于研制阶段。随着电子学、计算机科学、人工智能和机器人技术的进一步发展,功能完善的自主式空间机器人将在人类的活动中发挥巨大的作用。

目前,各国开发空间机器人都是根据对机器人的使用环境和使用方式的明确定义进行的,即这些空间机器人都是专用的特种机器人。因此,空间机器人还有如下多种分类方法。

3. 根据宇宙空间的位置划分

- (1) 在低轨道活动的机器人；
- (2) 在静止轨道活动的机器人；
- (3) 在月球表面活动的机器人；
- (4) 在行星(火星、金星和木星等)活动的机器人；
- (5) 在深宇宙活动的机器人。

4. 根据宇宙船的舱内外活动范围划分

- (1) 舱内活动机器人；
- (2) 舱外活动机器人。

5. 根据机器人的操作位置划分

- (1) 地上操作机器人:在地面站操作；
- (2) 舱内操作机器人:在宇宙船内部通过直视或操作台进行操作；
- (3) 舱外操作机器人:航天员穿宇宙服在舱外操作。

1.2 FFSR 的主要用途及研究意义

我国自 1970 年发射第一颗人造地球卫星以来,已经发射了几十颗卫星。尽管人造地球卫星已经取得了辉煌的成就,但它至今仍是航天信息的获取和传输工具,不能主动地完成某些工作。美国国家航空航天局(NASA)和日本国家空间发展局(NASDA)都提出了在卫星上搭接机械手和各种传感器,用它去完成在轨卫星服务(In-orbit Satellite Servicing),使卫星发挥更大的作用。

FFSR 是目前空间机器人领域的主要研究方向之一,它由机器人本体(卫星)和搭接在本体上的机械臂组成,它携带喷气装置,可在自由飞行或浮游状态下工作。近年来在国内外发表的文献中对这种具有移动功能的空间机器人有多种称谓,如机器人卫星(Robot Satellite)、空间机械手(Space Manipulator)等,实质上它们都是指在飞行器上搭载机械手的空间机器人系统,因此这些概念都是等价的。该种机器人的应用不但能提高作业效率,而且还能节省空间作业费用,因此它是将来能代替宇航员完成舱外作业的主要工具。

FFSR 的主要用途是:①对卫星做在轨服务,即用来对失效卫星捕捉、维修(包括零部件更换、能量补给等)后,重新投放使用,使卫星再生,或对捕捉卫星处理等作业。卫星的在轨服务,一方面能挽救失效卫星得到极大的经济效益,另一方面能克服由于卫星失效等因素造成的空间垃圾危害正常的卫星。②在未来星球大战中作为新式武器的俘获式卫星(Captured Satellite)——反卫星卫星,即用来俘获敌方卫星。这种反卫星对抗武器就是利用其自身探测设备发现敌方的卫星,对其进行跟踪并利用搭接在卫星上的空间机械手将它“捕捉”带回地面或空间站,获取其内藏的情报,这种反卫星航天器是第三代反卫星卫星,目前尚处于研究试验阶段。

由于 FFSR 工作在微重力的环境下,当机器人的手臂运动时,会对本体产生反作用力和力矩,从而改变机器人本体的位置和姿态,即 FFSR 的手臂和本体之间存在着运动学和动力学耦合。如果不考虑这种影响,而依然采用地面固定基座机器人的控制技术,FFSR 就无法完成预定的操作任务。为解决这个问题,必须找到能够考虑这种相互作用的建模方法和控制算法。为此,美国、日本等国家都在加紧进行 FFSR 的研制和实验工作。如美国斯

斯坦福大学空间机器人实验室在 1992 年就研制了 FFSR 地面试验平台;日本东芝公司和东京大学联合开发了自主型空间机器人操作实验平台等。另外,日本发射了 ETS-VII 型技术实验卫星,是一个机器人卫星,主要研究自主交会对接技术和面向 21 世纪人类空间活动所需的空间机器人技术。

未来的 FFSR,无论在操作灵巧程度,还是在对环境适应性方面,都会有一个大的进步。随着智能程度的提高,FFSR 将向着操作员监督下的智能自主操作方向发展。随着轨道飞行器的日益增多,在轨服务将成为一个极有发展前途的应用领域,这些服务包括损坏卫星的检修、损毁卫星的回收、燃料补充和设备更换等。参考日本发展无人自主交会对接技术和空间机器人的经验,尽快掌握轨道飞行器服务的关键技术,尤其是交会对接技术和空间机器人技术,发展有自己特色的空间新技术,无疑具有巨大的经济效益和社会效益。

1.3 FFSR 的研究现状

FFSR 的研究可以分为基本理论的研究、地面实验平台的研究和实际空间系统的开发等 3 个方面。

1.3.1 基本理论的研究

目前,对于 FFSR 基本理论的研究主要集中在以下几个方面。

1. 建模方法的研究

在 FFSR 完成任务的过程中需要准确地控制机械手运动,也就是如何保证机械手的末端效应器以指定的姿态到达指定的位置。有效的运动控制依赖于正确的建模、适当的控制方式及有效的运动规划算法。然而由于机器人的手臂运动会对本体产生反作用力,使 FFSR 的建模非常复杂。

FFSR 的建模主要有以下几种方法。

(1) Y. Umetani 和 K. Yoshida 通过对整个机器人系统引入动量守恒定律,提出了反映 FFSR 微运动学的广义雅可比矩阵(Generalized Jacobian Matrix, GJM)。R. Mukherjee 和 Y. Nakamura, K. Yamada 和 K. Tsuchiya, Y. Yokokohji 和 T. Toyoshima 等人采用不同的方法也都推导出 GJM。GJM 可应用到分解运动速度控制,转置雅可比控制和分解运动加速度控制等不同控制方法。

(2) Z. Vafa 和 S. Dubowsky 提出了虚拟机械手(Virtual Manipulator, VM)的概念用来描述 FFSR 的几何结构,VM 可用于简化 FFSR 系统的运动学和工作空间的计算,提供路径规划方法,并有助于分析、设计和控制 FFSR 系统。

(3) E. Papadopoulos 和 S. Dudowsky 提出了基于拉格朗日方程的动力学模型。它用来描述系统运动过程中速度、加速度和力矩的关系。

(4) 印度的 S. K. Saha 提出一个通用建模方法。为方便建模, S. K. Saha 提出了扩展质量矩阵(Extended Mass Matrix, EMM)和扩展总动量(Extended Total Momentum, ETM)等新概念,并结合动量定理和动能定理,得到一个描述系统运动动量的通用表示法。

2. 姿态控制的研究

在宇宙空间微重力环境下,搭载在机体上的机械手的运动会对机器人的本体产生反作用力,引起机器人的本体姿态发生变化。为保证机器人本体内携带的通信系统正常工作,必须保证本体上搭载的天线固定地指向某一位置,如地面通信站或轨道上的通信卫星等,因此,正确的姿态控制是非常必要的。

目前对姿态控制的研究可以分为三类:① 利用机器人本体内携带的反作用轮(Reaction Wheel)和反作用喷气装置(Reaction Thruster)固定本体的姿态或位置;② 设计特殊几何构造的机械手,利用对称性,平衡机械手对本体产生的反作用力,使机器人的本体姿态不变;③ 采用不同的控制算法,精确地控制机械手末端效应器的轨迹,使本体的姿态按预先指定的要求转动,或者维持最初姿态不变。

R. Longman 等人提出的姿态控制方案计算维持机器人本体姿态不变所需要的动量,并利用反作用轮提供这些动量。M. W. Walker 和 L. B. Wee 设计了一个自适应控制器,进行稳定的末端效应器的轨迹跟踪。K. Yoshida 提出的基于计算动量的反作用补偿方法,在进行姿态控制时,要比基于计算力矩的方法简单。这几种方法都是利用反作用喷气装置或反作用轮控制姿态。反作用喷气装置是一种外力控制装置,可以同时控制机器人的位置和姿态,它基于线动量守恒原理工作。喷气装置的使用将消耗机器人携带的不可再生的燃料,缩短机器人的轨道使用寿命,而且使用喷气装置会使机器人产生突然运动,不利于精确的操作。反作用轮是一种内力控制装置,它只能控制机器人本体的姿态,它基于角动量守恒原理工作。反作用轮使用通过太阳辐射生成的、并存储于可充电电池中的电能,这种能量是可再生的。但是充电电池提供的电能是有限的,因此反作用轮调节本体姿态的能力也有限。而且反作用轮在工作过程中很容易达到动量饱和,如果不将多余的能量转移,反作用轮就无法继续工作。因此,在姿态控制中应尽量少用反作用喷气装置和反作用轮。

E. Papadopoulos 和 A. Abu-Abed 提出,对于平面机械手,如果采用新的设计原则,会使机械手运动对本体产生的反动量为零。这时机器人的本体既不产生平动也不产生转动。为验证该理论,设计了一个 3 自由度 9 连杆的并行机械手,实验结果证明该方法是可行的。

目前多数的姿态控制算法仍集中在如何控制机械手的运动,使机器人本体的姿态不变或有规律地转动。Z. Vafa 和 S. Dubowsky 提出利用 FFSR 所特有的非完整冗余特性(Nonholonomic Redundancy),通过机械手在关节空间做闭合路径的运动,调节本体的姿态角。S. Dubowsky 和 M. A. Torres 已证明机械手按不同的方向运动在关节空间的方向是垂直的。D. Nenchev 提出的基于固定姿态限制雅可比矩阵(Fixed-Attitude-Restricted Jacobian Matrix, FARJM)的姿态控制方法,控制机械手的末端沿某一路径运动,同时使机器人的姿态按要求的方式改变,或使姿态改变最小。C. Fernandes 等人主要解决两种姿态控制问题。一是通过机械手的运动调整机器人本体的姿态,而机械手的初始姿态和终止姿态保持不变。二是机器人的本体姿态保持不变,但是改变机械手的姿态。S. K. Agrawal 和 S. Shirumalla 也提出了双臂协调的方案。M. Sampei 等人将控制理论应用到姿态控制中,提出一个新的基于时间状态控制表(Time-State Control Form)的方法。

3. 规划技术的研究

Dubowsky 和 Torres 等对 FFSR 动力学特点进行研究,提出一种空间机械臂的启发

式路径规划算法,用于降低姿态控制燃料的使用量。研究对机器人本体干扰值最小和最大的机械臂运动方向,指出产生最大和最小干扰值的机械臂运动在关节空间的方向是垂直的,并形成干扰图和增强干扰图,用于规划对本体干扰最小的机械臂运动。

Nakamura 和 Mukherjee 讨论了 FFSR 在自由浮游状态下的非完整和冗余性,利用 Liapunov 函数进行路径生成,同时控制机械臂的位姿和卫星本体的姿态。

Papadopulos 和 Dubowsky 在研究工作中发现 FFSR 在自由浮游状态下存在动力学奇异点。这些奇异点不能仅由运动学结构确定,而是依赖于系统的质量和惯量分布。动力学奇异点在笛卡儿空间是与路径相关的,在规划 FFSR 的运动时必须考虑这些奇异点。

Nenchev 等人在分析动量守恒方程的基础上,分析了在微重力环境下的冗余 FFSR 系统,并采用了伪逆冗余分解技术。一般的操作任务定义为末端效应器连续路径跟踪同时进行卫星姿态的控制。Nenchev 等人还提出一种固定姿态限制的雅可比矩阵。基于这个矩阵,通过数值积分计算生成的轨迹能保证机械臂的运动不对卫星的姿态产生影响。这个方法的一个优点是避开了机械臂的奇异点。

Vafa 和 Dubowsky 提出根据 FFSR 所固有的非完整冗余性,通过机械臂在关节空间做闭合路径的运动调节卫星本体的姿态角,Parlaktuna 等人也进行了类似的研究工作。

1.3.2 地面实验平台的研制

为了研究 FFSR 的运动学和动力学特性,以及验证各种关于 FFSR 的控制算法的可行性和有效性,仅仅做理论研究是不够的。建立 FFSR 的地面实验平台,利用它在地面上做模拟 FFSR 在空间环境下的各种仿真实验是一种有效的方法。FFSR 地面实验平台的一项关键技术就是微重力环境的模拟。目前,可以采用以下几种方法进行微重力环境的模拟。

(1) 在沿抛物线轨迹飞行的机舱内或在做自由落体运动的飞行器内做实验。在这种情况下,可以观测到真正的微重力环境,但这种实验的费用昂贵,而且微重力环境持续的时间也很短。德国的 ROTEX 空间机器人采用了这种方式做模拟实验。

(2) 在水池内做实验,这种方法要求水池内的水不带电,即为中性。航天员多采用这种方法训练,但水的阻力会影响航天员或机器人的运动,因此这种方法不能完全真正模拟空间微重力环境。

(3) 将实验模型用绳索悬挂起来,以抵消垂直重力的影响。采用这种方法的关键是消除振动和使悬挂系统结构简单。日本大阪大学和 Mitsubishi 公司等联合研制的数字仿真和伺服机械相结合的空间机器人地面实验平台仿真系统就采用这种方法。

(4) 利用空气的浮力在二维平面上以非接触形式支持模型漂浮在平台上。在各种模拟空间微重力的方法中,这种方法最简单。但它只能模拟二维平面的微重力环境。多数地面实验模型都采用这种方法,如美国的斯坦福大学、俄亥俄大学等研制的空间机器人实验模型都属于这种类型。

(5) 采用数学模型计算微重力环境下应产生的运动,然后借助外力使模型产生相应的运动。这种方法也称为“混合式”仿真,因为它将机械模型和数学模型结合起来。由于系统的反应要受到计算和伺服控制带宽的约束,在高频域的动力学实验将受到限制。美国的麻省理工学院研制的 VES- II 型空间机器人地面实验模型使用这种方法。

对于 FFSR 地面实验平台的研制,美国和日本做了许多卓有成效的工作。典型的实验平台包括:麻省理工学院 H. L. Alexander 等人研制的卫星机器人模拟车、美国俄亥俄大学 S. K. Agrawal 等人研制的双臂自由浮游装配机器人模型、麻省理工学院的 S. Dubowsky 建立的 PUMA560 空间机器人实验平台 VES- II 等、美国斯坦福大学空间机器人实验室研制的双臂 FFSR 地面实验平台、美国卡耐基梅隆大学机器人研究所开发的 SR 操作功能和控制结构的地面硬件仿真系统、日本东京大学 Y. Umetani 等人研制的二维自由飞行机器人卫星地面实验平台、日本东芝公司和东京大学联合开发的自主型 SR 操作实验平台 ASROT、日本空间系统实验室和富士实验室公司共同研制的高级冗余机械手 SR 实验平台 ASTRA、日本东北大学的 K. Yoshida 建立了自由浮游实验机器人卫星 EFFORTS- I 和 EFFORTS- II 型仿真模型、日本大阪大学和 Mitsubishi 公司建立了数字仿真与伺服机械相结合的 SR 地面仿真系统、日本电子实验室的 K. Machida 等人研制的参考航天员的自由飞行双臂遥控空间机器人地面实验平台,如图 1-5 所示。

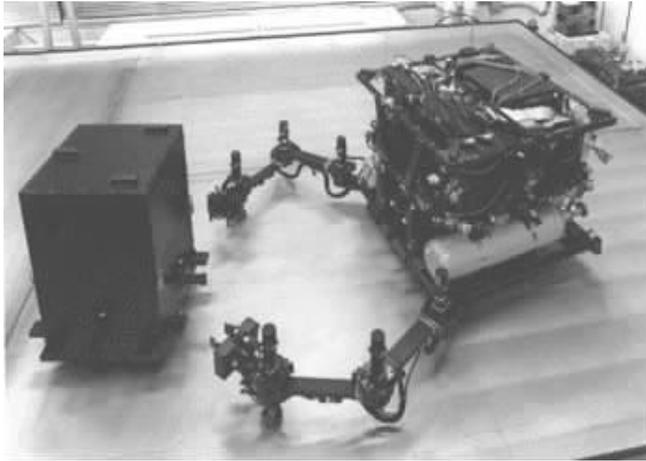


图 1-5 参考航天员空间机器人模型

1.3.3 空间试验系统的开发

日本 NASDA 于 1997 年 11 月 28 日成功地发射了一颗搭载有两个机械手的 ETS- VII 型近地轨道技术试验卫星。它的目的有两个:一是用于试验交会对接技术;二是试验机器人技术。如图 1-6 所示,ETS- VII 包括两颗卫星:追赶卫星(质量 2540kg)和目标卫星(质量 410kg)。

发射升空后,追赶卫星首先释放目标卫星,然后试验交会对接技术。ETS- VII 是世界上第一颗自由飞行空间机器人,为在实际的空间环境下进行空间机器人技术的试验提供了一个极好的机会。ETS- VII 的寿命为 1.5 年,目前已经成功地完成了全部实验任务,取得了预期的实验效果。ETS- VII 上的机器人实验系统模型如图 1-7 所示。FFSR 不但要求提供自主能力,而且还要求地面的安全可靠的遥控操作及对空间环境的适应性和卫星系统的协调性。

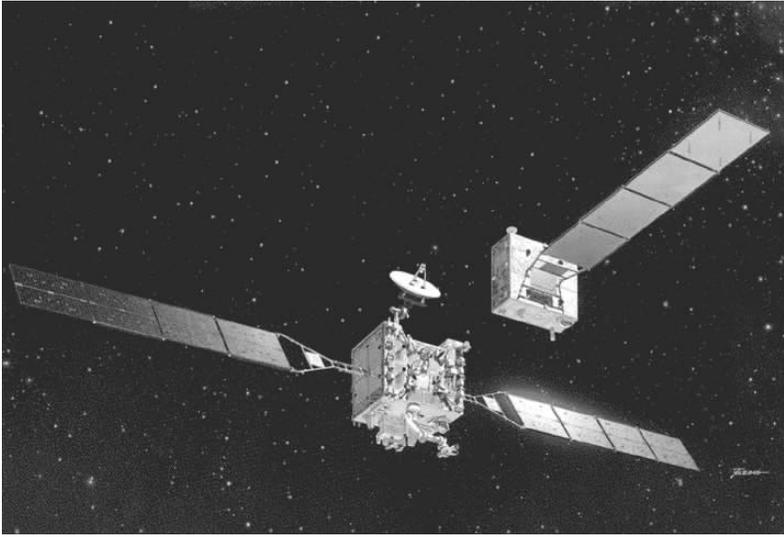


图 1-6 NASDA 的第七颗工程实验卫星

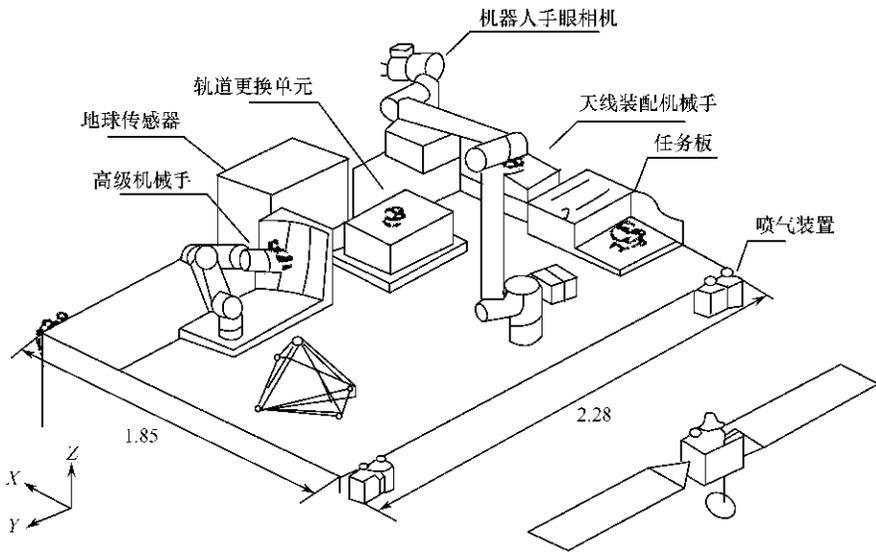


图 1-7 ETS-VII 上机器人实验系统模型

1.4 本书概要

本书介绍 FFSR 运动控制及仿真技术,着重阐述 FFSR 运动学模型和运动特性、运动规划方法、姿态控制方法以及 FFSR 地面实验平台。具体地说,本书包括下列内容。

(1) 阐述空间机器人的概念及分类,讨论 FFSR 的主要用途及其研究意义,介绍 FFSR 的研究现状。

(2) 给出双臂 FFSR 和多臂 FFSR 的运动学模型,讨论 FFSR 的运动学特性,阐述双

臂 FFSR 捕捉目标的运动规划方法。

(3) 讨论 FFSR 在空间微重力环境下的姿态控制方法,重点阐述基于受限最小干扰图的姿态控制、基于姿态稳定的冗余 FFSR 运动规划、基于姿态干扰预测的 FFSR 的姿态控制。

(4) 讨论基于关节驱动力矩的 FFSR 运动控制算法,阐述适用于 FFSR 的关节驱动力矩的求解算法和捕捉目标的力矩控制算法。

(5) 利用层次化方法构造模糊规则,提出基于模糊规则的 FFSR 运动控制方法。

(6) 研究 FFSR 飞行轨迹的优化问题,给出了单 FFSR 和多 FFSR 飞行轨迹的优化算法。

(7) 研究多个 FFSR 协调操作控制问题,给出多个 FFSR 协调操作的运动学和动力学模型以及协调控制算法。

(8) 介绍双臂 FFSR 地面试验平台,重点阐述各个组成模块,包括机器人模型、视觉系统、无线通信系统、网络系统、规划系统、气浮系统和操作台,并通过实验验证 FFSR 在空间微重力环境下的运动学特性。

第二章 FFSR 运动学模型及运动特性

FFSR工作在空间微重力环境下,它与地面机器人的不同之处在于地面机器人的基座是固定的,而 FFSR 的基座处于自由漂浮状态。机械手的运动会对机器人的本体产生反作用力和力矩,使本体的位置和姿态发生改变,因此传统的地面机器人的规划方法不再适用于 FFSR。在 FFSR 完成任务的过程中,需要精确地控制机械手移动其末端效应器以指定的姿态到达指定的位置,这种精确的控制依赖于对 FFSR 的正确建模。因此,建立 FFSR 在微重力环境中的运动学模型是非常重要的。本章从双臂自由飞行空间机器人(Dual Free -Flying Space Robot,DFFSR) 出发,再研究多臂 FFSR 的运动学模型和运动特性。

2.1 DFSR 的运动学模型

2.1.1 DFSR 的模型及参数定义

DFFSR 包括机器人的本体和本体上搭载的两只机械手。这两只机械手可有相同数目或不同数目的自由度。图 2-1 为双臂 FFSR 的模型。

图 2-1 中连杆 L_0 表示机器人的本体,连杆 L_i^l 表示左臂第 i 个连杆,连杆 L_j^r 表示右臂

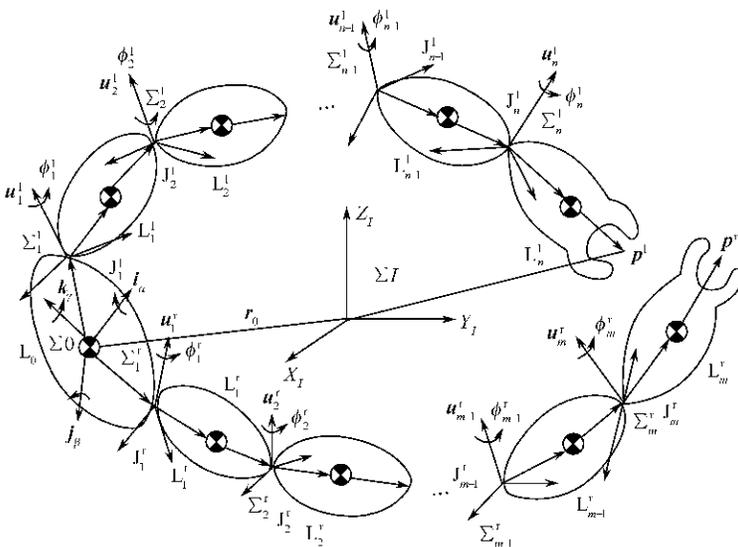


图 2-1 双臂 FFSR 的模型

第 j 个连杆。关节 J_i^l 连接左臂的连杆 $i-1$ 和连杆 j , 关节 J_j^r 连接右臂的连杆 $j-1$ 和连杆 j 。设左臂和右臂的转动关节数分别为 n 和 m , 则整个 FFSR 系统可看做为一个由 $n+m$ 个转动关节连接而成的 $n+m+1$ 个连杆系统。这里, 做如下假设:

(1) FFSR 的左臂和右臂的关节均为转动关节。

(2) FFSR 系统包括 $n+m$ 个关节和 $n+m+1$ 个连杆。其中每个关节都具有一个转动自由度, 并且该关节的速度是可控的, 而机器人本体的位置和姿态是不受控的。

(3) 在 FFSR 运动的初始时刻, 整个机器人系统在惯性坐标系中的位置和姿态都是已知的。

(4) 假设机械手的运动范围不受限制, 无任何外力和力矩作用于系统, 因此在运动过程中严格遵守动量守恒定律。

(5) 系统各个组成部分全部由刚体构成。

本章中用到的部分符号定义如下。

ΣI : 惯性坐标系。

$\Sigma 0$: 机器人本体坐标系, 固定于机器人本体质心。

$\Sigma_i^l(\Sigma_j^r)$: 左(右)臂连杆 $i(j)$ 坐标系, 固定于左(右)臂关节 $i(j)$, 其 z 轴与关节 $i(j)$ 的转动轴一致。其中 $i=1, 2, \dots, n; j=1, 2, \dots, m$ 。当 $i=n$ 时为左臂末端连杆坐标系, 当 $j=m$ 时为右臂末端连杆坐标系。

$u_j^l(u_j^r) \in R^3$: 左(右)臂连杆 $i(j)$ 坐标系中与 z 轴方向一致的单位向量, 它为左(右)臂关节 $i(j)$ 的旋转轴(单位: m)。

$m_i^l(m_j^r)$: 左(右)臂连杆 $i(j)$ 的质量(单位: kg)。

W : 系统的总质量(单位: kg)。

${}^1r_i^l({}^1r_j^r) \in R^3$: 左(右)臂连杆 $i(j)$ 的质心在惯性坐标系中的位置向量(单位: m)。

${}^1p_i^l({}^1p_j^r) \in R^3$: 左(右)臂末端效应器在惯性坐标系中的位置向量(单位: m)。

${}^i a_i^l({}^j a_j^r) \in R^3$: 在左(右)臂连杆坐标系 $i(j)$ 中, 从关节 $i(j)$ 指向连杆 $i(j)$ 质心的位置向量(单位: m)。

${}^i b_i^l({}^j b_j^r) \in R^3$: 在左(右)臂连杆坐标系 $i(j)$ 中, 从连杆 $i(j)$ 质心指向关节 $i+1(j+1)$ 的位置向量(单位: m)。

${}^i l_i^l({}^j l_j^r) \in R^3$: 在左(右)臂连杆坐标系 $i(j)$ 中, 从关节 $i(j)$ 指向关节 $i+1(j+1)$ 的位置向量(单位: m)。

${}^i I_i^l({}^j I_j^r) \in R^3$: 在左(右)臂连杆坐标系 $i(j)$ 中, 连杆 $i(j)$ 相对其质心的惯性矩阵。它为一个常数矩阵(单位: $\text{kg} \cdot \text{m}^2 \cdot \text{m}$)。

${}^1 A_i^l({}^1 A_j^r) \in R^{3 \times 3}$: 左(右)臂连杆坐标系 $i(j)$ 到惯性坐标系的旋转矩阵($\text{kg} \cdot \text{m}^2$)。

$\phi_i^l(\phi_j^r)$: 左(右)臂关节 $i(j)$ 的旋转角(单位: rad)。

α, β, γ : 机器人本体的姿态角(单位: rad)。

在上面定义中各连杆坐标系都根据 Denavit-Hartenberg 惯例定义。

2.1.2 运动学方程

对于一个 DFFSR 系统, 它的系统质心可表示为:

$$m_0 \mathbf{r}_0 + \sum_{i=1}^n m_i^l \mathbf{r}_i^l + \sum_{i=1}^m m_i^r \mathbf{r}_i^r = \mathbf{r}_G (m_0 + \sum_{i=1}^n m_i^l + \sum_{i=1}^m m_i^r) \quad (2-1)$$

式中, $\mathbf{r}_G \in R^3$ 为惯性坐标系中系统质心的位置向量; m_0 为机器人本体的质量; \mathbf{r}_0 为惯性坐标系中本体质心的位置向量。

由于系统在运动过程中不受外力和力矩的作用, 因此系统满足线动量和角动量守恒定律, 其形式如下:

$$m_0^l \dot{\mathbf{r}}_0 + \sum_{i=1}^n m_i^l \dot{\mathbf{r}}_i^l + \sum_{i=1}^m m_i^r \dot{\mathbf{r}}_i^r = 0 \quad (2-2)$$

$${}^1 I_0^l \boldsymbol{\omega}_0 + m_0^l \mathbf{r}_0 \times \dot{\mathbf{r}}_0 + \sum_{i=1}^n ({}^1 I_i^l \boldsymbol{\omega}_i + m_i^l \mathbf{r}_i^l \times \dot{\mathbf{r}}_i^l) + \sum_{i=1}^m ({}^1 I_i^r \boldsymbol{\omega}_i^r + m_i^r \mathbf{r}_i^r \times \dot{\mathbf{r}}_i^r) = 0 \quad (2-3)$$

由各连杆的几何关系可得:

$${}^1 \mathbf{r}_i^l - {}^1 \mathbf{r}_{i-1}^l = {}^i \mathbf{a}_i^l + {}^{i-1} \mathbf{b}_{i-1}^l \quad (2-4)$$

$${}^1 \mathbf{r}_i^r - {}^1 \mathbf{r}_{i-1}^r = {}^i \mathbf{a}_i^r + {}^{i-1} \mathbf{b}_{i-1}^r \quad (2-5)$$

$${}^1 \mathbf{p}^l = {}^1 \mathbf{r}_0 + {}^0 \mathbf{b}_0^l + \sum_{i=1}^n \mathbf{l}_i^l \quad (2-6)$$

$${}^1 \mathbf{p}^r = {}^1 \mathbf{r}_0 + {}^0 \mathbf{b}_0^r + \sum_{i=1}^m \mathbf{l}_i^r \quad (2-7)$$

本章所研究的 DFFSR 系统为双臂 6 自由度 FFSR 系统, 该机器人由机器人本体和对称的左臂和右臂组成, 每一机械臂包括 3 个转动关节, 机器人的本体包括一个固定臂。该机器人的模型如图 2-2 所示。

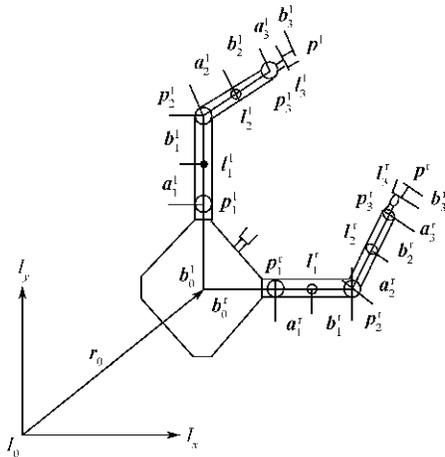


图 2-2 DFFSR 系统几何模型

根据系统质心不变原理,该机器人的系统质心为:

$$m_0 \mathbf{r}_0 + \sum_{i=1}^3 m_i^l \mathbf{r}_i^l + \sum_{i=1}^3 m_i^r \mathbf{r}_i^r = 0 \quad (2-8)$$

又由各连杆的几何关系可得:

$$\mathbf{r}_i^r - \mathbf{r}_{i-1}^r = {}^i \mathbf{a}_i^r + {}^{i-1} \mathbf{b}_{i-1}^r \quad (2-9)$$

$$\mathbf{r}_i^l - \mathbf{r}_{i-1}^l = {}^i \mathbf{a}_i^l + {}^{i-1} \mathbf{b}_{i-1}^l \quad (2-10)$$

该机器人系统的特征方程为:

$$\mathbf{p}^r = \mathbf{r}_0 + {}^0 \mathbf{b}_0^r + \sum_{i=1}^3 {}^i \mathbf{l}_i^r \quad (2-11)$$

$$\mathbf{p}^l = \mathbf{r}_0 + {}^0 \mathbf{b}_0^l + \sum_{i=1}^3 {}^i \mathbf{l}_i^l \quad (2-12)$$

系统的总质量为:

$$W = m_0 + \sum_{i=1}^3 m_i^l + \sum_{i=1}^3 m_i^r \quad (2-13)$$

由式(2-8) ~ 式(2-13) 可得:

$$\mathbf{p}^r = \begin{bmatrix} \mathbf{p}_x^r \\ \mathbf{p}_y^r \end{bmatrix} = \sum_{i=0}^7 \mathbf{k}_i^r \begin{bmatrix} C_i \\ S_i \end{bmatrix} \quad (2-14)$$

$$\mathbf{p}^l = \begin{bmatrix} \mathbf{p}_x^l \\ \mathbf{p}_y^l \end{bmatrix} = \sum_{i=0}^7 \mathbf{k}_i^l \begin{bmatrix} C_i \\ S_i \end{bmatrix} \quad (2-15)$$

式中

$$\begin{aligned} \mathbf{k}_0^r &= \frac{(m_0 + m_1^l + m_2^l + m_3^l) \mathbf{b}_0^r}{W} \\ \mathbf{k}_1^r &= \frac{m_1^r \mathbf{b}_1^r + (m_0 + m_1^l + m_2^l + m_3^l) \mathbf{l}_1^r}{W} \\ \mathbf{k}_2^r &= \frac{m_2^r \mathbf{b}_2^r + (m_0 + m_1^r + m_1^l + m_2^l + m_3^l) \mathbf{l}_2^r}{W} \\ \mathbf{k}_3^r &= \frac{m_3^r \mathbf{b}_3^r + (m_0 + m_1^r + m_2^r + m_1^l + m_2^l + m_3^l) \mathbf{l}_3^r}{W} \\ \mathbf{k}_4^r &= -\frac{m_1^l \mathbf{a}_1^l + (m_2^l + m_3^l) \mathbf{l}_1^l}{W} \\ \mathbf{k}_5^r &= -\frac{m_2^l \mathbf{a}_2^l + m_3^l \mathbf{l}_2^l}{W} \end{aligned}$$

$$\begin{aligned}
\mathbf{k}_6^r &= -\frac{m_3^1 \mathbf{a}_3^1}{W} \\
\mathbf{k}_7^r &= -\frac{(m_1^1 + m_2^1 + m_3^1) \mathbf{b}_0^1}{W} \\
\mathbf{k}_0^1 &= -\frac{(m_1^r + m_2^r + m_3^r) \mathbf{b}_0^r}{W} \\
\mathbf{k}_1^1 &= -\frac{m_1^r \mathbf{a}_1^r + (m_2^r + m_3^r) \mathbf{l}_1^r}{W} \\
\mathbf{k}_2^1 &= -\frac{m_2^r \mathbf{a}_2^r + m_3^r \mathbf{l}_2^r}{W} \\
\mathbf{k}_3^1 &= -\frac{m_3^r \mathbf{a}_3^r}{W} \\
\mathbf{k}_4^1 &= \frac{(m_0 + m_1^r + m_2^r + m_3^r) \mathbf{l}_1^1 + m_1^1 \mathbf{b}_1^1}{W} \\
\mathbf{k}_5^1 &= \frac{(m_0 + m_1^r + m_2^r + m_3^r + m_1^1) \mathbf{l}_2^1 + m_2^1 \mathbf{b}_2^1}{W} \\
\mathbf{k}_6^1 &= \frac{(m_0 + m_1^r + m_2^r + m_3^r + m_1^1 + m_2^1) \mathbf{l}_3^1 + m_3^1 \mathbf{b}_3^1}{W} \\
\mathbf{k}_7^1 &= \frac{(m_0 + m_1^r + m_2^r + m_3^r) \mathbf{b}_0^1}{W}
\end{aligned}$$

$$C_i = \cos\left(\phi_0 + \sum_{k=1}^i \phi_k^r - \frac{\pi}{4}\right) \quad S_i = \sin\left(\phi_0 + \sum_{k=1}^i \phi_k^r - \frac{\pi}{4}\right) \quad (0 \leq i \leq 3)$$

$$C_i = \cos\left(\phi_0 + \sum_{k=4}^i \phi_{k-3}^1 + \frac{\pi}{4}\right) \quad S_i = \sin\left(\phi_0 + \sum_{k=4}^i \phi_{k-3}^1 + \frac{\pi}{4}\right) \quad (4 \leq i \leq 6)$$

$$C_i = \cos\left(\phi_0 + \frac{\pi}{4}\right) \quad S_i = \sin\left(\phi_0 + \frac{\pi}{4}\right) \quad (i = 7)$$

将式(2-14)和式(2-15)对时间微分,可得:

$$\begin{aligned}
\dot{\mathbf{p}}^r &= \sum_{i=0}^7 \mathbf{k}_i^r \begin{bmatrix} -S_i \\ C_i \end{bmatrix} \dot{\phi}_0 + \\
&\begin{bmatrix} -k_1^r S_1 - k_2^r S_2 - k_3^r S_3 - k_2^r S_2 - k_3^r S_3 - k_3^r S_3 - k_4^r S_4 - k_5^r S_5 - k_6^r S_6 - k_5^r S_5 - k_6^r S_6 - k_6^r S_6 \\ k_1^r C_1 + k_2^r C_2 + k_3^r C_3 \quad k_2^r C_2 + k_3^r C_3 \quad k_3^r C_3 \quad k_4^r C_4 + k_5^r C_5 + k_6^r C_6 \quad k_5^r C_5 + k_6^r C_6 \quad k_6^r C_6 \end{bmatrix} \cdot \\
&[\dot{\phi}_1^r, \dot{\phi}_2^r, \dot{\phi}_3^r, \dot{\phi}_1^1, \dot{\phi}_2^1, \dot{\phi}_3^1]^T
\end{aligned} \tag{2-16}$$

$$\dot{\mathbf{p}}^1 = \sum_{i=0}^7 \mathbf{k}_i^1 \begin{bmatrix} -S_i \\ C_i \end{bmatrix} \dot{\phi}_0 +$$

$$\begin{bmatrix} -\mathbf{k}_1^1 S_1 - \mathbf{k}_2^1 S_2 - \mathbf{k}_3^1 S_3 - \mathbf{k}_2^1 S_2 - \mathbf{k}_3^1 S_3 - \mathbf{k}_3^1 S_3 - \mathbf{k}_4^1 S_4 - \mathbf{k}_5^1 S_5 - \mathbf{k}_6^1 S_6 - \mathbf{k}_5^1 S_5 - \mathbf{k}_6^1 S_6 - \mathbf{k}_6^1 S_6 \\ \mathbf{k}_1^1 C_1 + \mathbf{k}_2^1 C_2 + \mathbf{k}_3^1 C_3 & \mathbf{k}_2^1 C_2 + \mathbf{k}_3^1 C_3 & \mathbf{k}_3^1 C_3 & \mathbf{k}_4^1 C_4 + \mathbf{k}_5^1 C_5 + \mathbf{k}_6^1 C_6 & \mathbf{k}_5^1 C_5 + \mathbf{k}_6^1 C_6 & \mathbf{k}_6^1 C_6 \end{bmatrix} \cdot$$

$$[\dot{\phi}_1^r, \dot{\phi}_2^r, \dot{\phi}_3^r, \dot{\phi}_1^l, \dot{\phi}_2^l, \dot{\phi}_3^l]^T \quad (2-17)$$

将式(2-16)和式(2-17)联立,可得到下式:

$$\dot{\mathbf{p}} = \begin{bmatrix} \dot{\mathbf{p}}^r \\ \dot{\mathbf{p}}^l \end{bmatrix} = \mathbf{J}_S \dot{\phi}_S + \mathbf{J}_M \dot{\phi}_M \quad (2-18)$$

由式(2-9)和式(2-10)可得:

$$\mathbf{r}_1^r = \mathbf{r}_0 + \mathbf{b}_0^r + \mathbf{a}_1^r \quad (2-19a)$$

$$\mathbf{r}_2^r = \mathbf{r}_0 + \mathbf{b}_0^r + \mathbf{l}_1^r + \mathbf{a}_2^r \quad (2-19b)$$

$$\mathbf{r}_3^r = \mathbf{r}_0 + \mathbf{b}_0^r + \mathbf{l}_1^r + \mathbf{l}_2^r + \mathbf{a}_3^r \quad (2-19c)$$

$$\mathbf{r}_1^l = \mathbf{r}_0 + \mathbf{b}_0^l + \mathbf{a}_1^l \quad (2-19d)$$

$$\mathbf{r}_2^l = \mathbf{r}_0 + \mathbf{b}_0^l + \mathbf{l}_1^l + \mathbf{a}_2^l \quad (2-19e)$$

$$\mathbf{r}_3^l = \mathbf{r}_0 + \mathbf{b}_0^l + \mathbf{l}_1^l + \mathbf{l}_2^l + \mathbf{a}_3^l \quad (2-19f)$$

将式(2-19a) ~ (2-19f)代入式(2-8)中,可得到下式:

$$\mathbf{r}_0 = \sum_{k=0}^7 X_{0k} \begin{bmatrix} C_k \\ S_k \end{bmatrix} \quad (2-20a)$$

$$\mathbf{r}_i^r = \sum_{k=0}^7 Y_{ik} \begin{bmatrix} C_k \\ S_k \end{bmatrix} \quad (i = 1, 2, 3) \quad (2-20b)$$

$$\mathbf{r}_i^l = \sum_{k=0}^7 Z_{ik} \begin{bmatrix} C_k \\ S_k \end{bmatrix} \quad (i = 1, 2, 3) \quad (2-20c)$$

式中, X_{0k} 、 Y_{ik} 和 Z_{ik} 为与质量和长度有关的常数。将式(2-20a) ~ 式(2-20c)对时间微分,则有:

$$\dot{\mathbf{r}}_0 = \sum_{k=0}^7 X_{0k} \begin{bmatrix} -S_i \\ C_i \end{bmatrix} \dot{\phi}_0 +$$

$$\begin{bmatrix} -X_{01} S_1 - X_{02} S_2 - X_{03} S_3 - X_{02} S_2 - X_{03} S_3 - X_{03} S_3 - X_{04} S_4 - X_{05} S_5 - X_{06} S_6 - X_{05} S_5 - X_{06} S_6 - X_{06} S_6 \\ X_{01} C_1 + X_{02} C_2 + X_{03} C_3 & X_{02} C_2 + X_{03} C_3 & X_{03} C_3 & X_{04} C_4 + X_{05} C_5 + X_{06} C_6 & X_{05} C_5 + X_{06} C_6 & X_{06} C_6 \end{bmatrix} \cdot$$

$$[\dot{\phi}_1^r, \dot{\phi}_2^r, \dot{\phi}_3^r, \dot{\phi}_1^l, \dot{\phi}_2^l, \dot{\phi}_3^l]^T \quad (2-21a)$$

$$\dot{\mathbf{r}}_i^r = \sum_{k=0}^7 Y_{ik} \begin{bmatrix} -S_i \\ C_i \end{bmatrix} \dot{\phi}_0 + \begin{bmatrix} -Y_{i1}S_1 - Y_{i2}S_2 - Y_{i3}S_3 - Y_{i2}S_2 - Y_{i3}S_3 - Y_{i3}S_3 - Y_{i4}S_4 - Y_{i5}S_5 - Y_{i6}S_6 - Y_{i5}S_5 - Y_{i6}S_6 - Y_{i6}S_6 \\ Y_{i1}C_1 + Y_{i2}C_2 + Y_{i3}C_3 & Y_{i2}C_2 + Y_{i3}C_3 & Y_{i3}C_3 & Y_{i4}C_4 + Y_{i5}C_5 + Y_{i6}C_6 & Y_{i5}C_5 + Y_{i6}C_6 & Y_{i6}C_6 \end{bmatrix} \cdot [\dot{\phi}_1^r, \dot{\phi}_2^r, \dot{\phi}_3^r, \dot{\phi}_1^l, \dot{\phi}_2^l, \dot{\phi}_3^l]^T \quad (i = 1, 2, 3) \quad (2-21b)$$

$$\dot{\mathbf{r}}_i^l = \sum_{k=0}^7 Z_{ik} \begin{bmatrix} -S_i \\ C_i \end{bmatrix} \dot{\phi}_0 + \begin{bmatrix} -Z_{i1}S_1 - Z_{i2}S_2 - Z_{i3}S_3 - Z_{i2}S_2 - Z_{i3}S_3 - Z_{i3}S_3 - Z_{i4}S_4 - Z_{i5}S_5 - Z_{i6}S_6 - Z_{i5}S_5 - Z_{i6}S_6 - Z_{i6}S_6 \\ Z_{i1}C_1 + Z_{i2}C_2 + Z_{i3}C_3 & Z_{i2}C_2 + Z_{i3}C_3 & Z_{i3}C_3 & Z_{i4}C_4 + Z_{i5}C_5 + Z_{i6}C_6 & Z_{i5}C_5 + Z_{i6}C_6 & Z_{i6}C_6 \end{bmatrix} \cdot [\dot{\phi}_1^r, \dot{\phi}_2^r, \dot{\phi}_3^r, \dot{\phi}_1^l, \dot{\phi}_2^l, \dot{\phi}_3^l]^T \quad (i = 1, 2, 3) \quad (2-21c)$$

由式(2-20a) ~ 式(2-20c) 和式(2-21a) ~ 式(2-21c) 得:

$$\mathbf{r}_0 \times \dot{\mathbf{r}}_0 = H_{S0} \dot{\phi}_0 + H_{M01} \dot{\phi}_1^r + H_{M02} \dot{\phi}_2^r + H_{M03} \dot{\phi}_3^r + H_{M04} \dot{\phi}_1^l + H_{M05} \dot{\phi}_2^l + H_{M06} \dot{\phi}_3^l \quad (2-22a)$$

$$\mathbf{r}_i^r \times \dot{\mathbf{r}}_i^r = H_{S_i^r}^r \dot{\phi}_0 + H_{M_{i1}^r}^r \dot{\phi}_1^r + H_{M_{i2}^r}^r \dot{\phi}_2^r + H_{M_{i3}^r}^r \dot{\phi}_3^r + H_{M_{i4}^r}^r \dot{\phi}_1^l + H_{M_{i5}^r}^r \dot{\phi}_2^l + H_{M_{i6}^r}^r \dot{\phi}_3^l \quad (2-22b)$$

$$\mathbf{r}_i^l \times \dot{\mathbf{r}}_i^l = H_{S_i^l}^l \dot{\phi}_0 + H_{M_{i1}^l}^l \dot{\phi}_1^r + H_{M_{i2}^l}^l \dot{\phi}_2^r + H_{M_{i3}^l}^l \dot{\phi}_3^r + H_{M_{i4}^l}^l \dot{\phi}_1^l + H_{M_{i5}^l}^l \dot{\phi}_2^l + H_{M_{i6}^l}^l \dot{\phi}_3^l \quad (2-22c)$$

令:

$$\boldsymbol{\omega}_{0j} = X_{00} X_{0j} \cos\left(\sum_{k=1}^j \phi_k^r\right) \quad (j = 1, 2, 3)$$

$$s_{0j}^i = Q_{i0} Q_{ij} \cos\left(\sum_{k=1}^j \phi_k^r\right) \quad (i, j = 1, 2, 3)$$

$$\boldsymbol{\omega}_{0j} = -X_{00} X_{0j} \sin\left(\sum_{k=4}^j \phi_{k-3}^l\right) \quad (j = 4, 5, 6)$$

$$s_{0j}^i = -Q_{i0} Q_{ij} \sin\left(\sum_{k=4}^j \phi_{k-3}^l\right) \quad (i = 1, 2, 3; j = 4, 5, 6)$$

$$\boldsymbol{\omega}_{1j} = X_{01} X_{0j} \cos\left(\sum_{k=2}^j \phi_k^r\right) \quad (j = 2, 3)$$

$$s_{1j}^i = Q_{i1} Q_{ij} \cos\left(\sum_{k=2}^j \phi_k^r\right) \quad (i = 1, 2, 3; j = 2, 3)$$

$$\boldsymbol{\omega}_{1j} = -X_{01}X_{0j} \sin\left(\sum_{k=4}^j \phi_{k-3}^1 - \phi_1^r\right) \quad (j = 4, 5, 6)$$

$$\boldsymbol{s}_{1j}^i = -Q_{i1}Q_{ij} \sin\left(\sum_{k=4}^j \phi_{k-3}^1 - \phi_1^r\right) \quad (i = 1, 2, 3; j = 4, 5, 6)$$

$$\boldsymbol{\omega}_{17} = X_{01}X_{07} \sin(\phi_1^r)$$

$$\boldsymbol{s}_{17}^i = Q_{i1}Q_{i7} \sin(\phi_1^r)$$

$$\boldsymbol{\omega}_{23} = X_{02}X_{03} \cos(\phi_3^r)$$

$$\boldsymbol{s}_{23}^i = Q_{i2}Q_{i3} \cos(\phi_3^r)$$

$$\boldsymbol{\omega}_{2j} = -X_{02}X_{0j} \sin\left(\sum_{k=4}^j \phi_{k-3}^1 - \phi_1^r - \phi_2^r\right) \quad (j = 4, 5, 6)$$

$$\boldsymbol{s}_{2j}^i = -Q_{i2}Q_{ij} \sin\left(\sum_{k=4}^j \phi_{k-3}^1 - \phi_1^r - \phi_2^r\right) \quad (i = 1, 2, 3; j = 4, 5, 6)$$

$$\boldsymbol{\omega}_{27} = X_{02}X_{07} \sin\left(\sum_{k=1}^2 \phi_k^r\right)$$

$$\boldsymbol{s}_{27}^i = Q_{i2}Q_{i7} \sin\left(\sum_{k=1}^2 \phi_k^r\right)$$

$$\boldsymbol{\omega}_{3j} = -X_{03}X_{0j} \sin\left(\sum_{k=4}^j \phi_{k-3}^1 - \sum_{k=1}^3 \phi_k^r\right) \quad (j = 4, 5, 6)$$

$$\boldsymbol{s}_{3j}^i = -Q_{i3}Q_{ij} \sin\left(\sum_{k=4}^j \phi_{k-3}^1 - \sum_{k=1}^3 \phi_k^r\right) \quad (i = 1, 2, 3; j = 4, 5, 6)$$

$$\boldsymbol{\omega}_{37} = X_{03}X_{07} \sin\left(\sum_{k=1}^3 \phi_k^r\right)$$

$$\boldsymbol{s}_{37}^i = Q_{i3}Q_{i7} \sin\left(\sum_{k=1}^3 \phi_k^r\right)$$

$$\boldsymbol{\omega}_{4j} = X_{04}X_{0j} \cos\left(\sum_{k=5}^j \phi_{k-3}^1\right) \quad (j = 5, 6)$$

$$\boldsymbol{s}_{4j}^i = Q_{i4}Q_{ij} \cos\left(\sum_{k=5}^j \phi_{k-3}^1\right) \quad (i = 1, 2, 3; j = 5, 6)$$

$$\boldsymbol{\omega}_{56} = X_{05}X_{06} \cos(\phi_3^1)$$

$$\boldsymbol{s}_{56}^i = Q_{i5}Q_{i6} \cos(\phi_3^1)$$

$$\boldsymbol{\omega}_{7j} = -X_{0j}X_{07} \cos\left(\sum_{k=4}^j \phi_{k-3}^1\right) \quad (j = 4, 5, 6)$$

$$s_{j7}^i = Q_{ij} Q_{i7} \cos\left(\sum_{k=4}^j \phi_{k-3}^1\right) \quad (j = 4, 5, 6)$$

上面各式中,当 $s_{jk}^i = u_{jk}^i$ 时,有 $Q_j = Y_j$,当 $s_{jk}^i = v_{jk}^i$ 时,则有 $Q_j = Z_j$ 。而在式(2-22a)、式(2-22b)和式(2-22c)中,

$$H_{M01} = \sum_{j=1}^3 (X_{0j}^2 + \omega_{0j}) + \sum_{j=1}^3 \sum_{k=j+1}^7 \omega_{jk} + \omega_{12} + \omega_{13} + \omega_{23}$$

$$H_{Mi1}^c = \sum_{j=1}^3 (Q_{ij}^2 + s_{0j}^i) + \sum_{j=1}^3 \sum_{k=j+1}^7 s_{jk}^i + s_{12}^i + s_{13}^i + s_{23}^i$$

$$H_{M02} = \sum_{j=2}^3 (X_{0j}^2 + \omega_{0j}) + \sum_{j=2}^3 \sum_{k=j+1}^7 \omega_{jk} + \omega_{12} + \omega_{13} + \omega_{23}$$

$$H_{Mi2}^c = \sum_{j=2}^3 (Q_{ij}^2 + s_{0j}^i) + \sum_{j=2}^3 \sum_{k=j+1}^7 s_{jk}^i + s_{12}^i + s_{13}^i + s_{23}^i$$

$$H_{M03} = X_{03}^2 + \sum_{j=0}^2 \omega_{j3} + \sum_{j=4}^7 \omega_{3j}$$

$$H_{Mi3}^c = Q_{i3}^2 + \sum_{j=0}^2 s_{j3}^i + \sum_{j=4}^7 s_{3j}^i$$

$$H_{M04} = \sum_{j=4}^6 (X_{0j}^2 + \omega_{0j}) + \sum_{k=4}^6 \sum_{j=1}^{k-1} \omega_{jk} + \sum_{k=4}^6 \sum_{j=k+1}^7 \omega_{kj}$$

$$H_{Mi4}^c = \sum_{j=4}^6 (Q_{ij}^2 + s_{0j}^i) + \sum_{k=4}^6 \sum_{j=1}^{k-1} s_{jk}^i + \sum_{k=4}^6 \sum_{j=k+1}^7 s_{kj}^i$$

$$H_{M05} = \sum_{j=5}^6 (X_{0j}^2 + \omega_{0j}) + \sum_{k=5}^6 \sum_{j=1}^{k-1} \omega_{jk} + \sum_{k=5}^6 \sum_{j=k+1}^7 \omega_{kj}$$

$$H_{Mi5}^c = \sum_{j=5}^6 (Q_{ij}^2 + s_{0j}^i) + \sum_{k=5}^6 \sum_{j=1}^{k-1} s_{jk}^i + \sum_{k=5}^6 \sum_{j=k+1}^7 s_{kj}^i$$

$$H_{M06} = X_{06}^2 + \sum_{j=0}^5 \omega_{j6} + \sum_{j=4}^7 \omega_{67}$$

$$H_{Mi6}^c = Q_{i6}^2 + \sum_{j=0}^5 s_{j6}^i + s_{67}^i$$

在上面各式中,当 $i = 1, 2, 3$ 时,

$$H_j^c = H_j^i, Q_j = Y_j, s_{jk}^i = u_{jk}^i$$

当 $i = 4, 5, 6$ 时,

$$H_j^c = H_j^i, Q_j = Z_j, s_{jk}^i = v_{jk}^i$$

将式(2-22a)、式(2-22b)和式(2-22c)代入角动量守恒定律,即

$$I_0 \boldsymbol{\omega}_0 + m_0 \mathbf{r}_0 \times \dot{\mathbf{r}}_0 + \sum_{i=1}^3 (I_i^r \boldsymbol{\omega}_i^r + m_i^r \mathbf{r}_i^r \times \dot{\mathbf{r}}_i^r) + \sum_{i=1}^3 (I_i^l \boldsymbol{\omega}_i^l + m_i^l \mathbf{r}_i^l \times \dot{\mathbf{r}}_i^l) = 0 \quad (2-23)$$

可得:

$$I_S \dot{\boldsymbol{\phi}}_S + \mathbf{I}_M \dot{\boldsymbol{\phi}}_M = 0 \quad (2-24)$$

式中

$$\begin{aligned} I_S &= I_0 + \sum_{i=1}^3 (I_i^r + m_i^r H_{S_i}^r) + \sum_{i=1}^3 (I_i^l + m_i^l H_{S_i}^l) \\ \mathbf{I}_M &= [I_{M1}^r, I_{M2}^r, I_{M3}^r, I_{M1}^l, I_{M2}^l, I_{M3}^l] \\ I_{Mi}^r &= \sum_{j=i}^3 I_j^r + \sum_{j=1}^3 m_j^r H_{M_{ji}}^r + \sum_{j=1}^3 m_j^l H_{M_{ji}}^l \quad (i = 1, 2, 3) \\ I_{Mi}^l &= \sum_{j=i}^3 I_j^l + \sum_{j=1}^3 m_j^r H_{M_{j(i+3)}}^r + \sum_{j=1}^3 m_j^l H_{M_{j(i+3)}}^l \quad (i = 1, 2, 3) \end{aligned}$$

将式(2-24)代入式(2-18)中,可得到下式:

$$\dot{\mathbf{p}} = \begin{bmatrix} \dot{\mathbf{p}}^r \\ \dot{\mathbf{p}}^l \end{bmatrix} = (\mathbf{J}_M - \mathbf{J}_S \mathbf{I}_S^{-1} \mathbf{I}_M) \dot{\boldsymbol{\phi}}_M = \mathbf{J}^* \dot{\boldsymbol{\phi}}_M \quad (2-25)$$

式中, $\mathbf{J}^* \in \mathbf{R}^{4 \times 6}$ 为双臂 6 自由度 FFSR 的广义雅可比矩阵,它描述了机械手末端运动速度与各关节角速度之间的关系; $\dot{\boldsymbol{\phi}}_M \in \mathbf{R}^{6 \times 1}$ 为关节角速度矩阵。设 $(\mathbf{J}^*)^+$ 为 \mathbf{J}^* 的广义逆,则式(2-25)可写为:

$$\dot{\boldsymbol{\phi}}_M = (\mathbf{J}^*)^+ \dot{\mathbf{p}} \quad (2-26)$$

式(2-26)将末端运动速度分解成各关节角速度,根据 $\dot{\boldsymbol{\phi}}_M$ 和 $\dot{\mathbf{p}}$ 之间存在的对应关系,可以将地面机器人的控制算法应用在空间机器人中。当 \mathbf{J}^* 的广义逆不存在时,则机器人处在奇异位置。若 $\mathbf{J}^* = \mathbf{BC}$ 是 \mathbf{J}^* 的最大秩分解,则

$$(\mathbf{J}^*)^+ = \mathbf{C}^T (\mathbf{C} \mathbf{C}^T)^{-1} (\mathbf{B}^T \mathbf{B})^{-1} \mathbf{B}^T \quad (2-27)$$

2.2 DFFSR 通用运动学模型

上节所建立的运动学模型是以机器人本体为参考体,但也可以选择其他参考体。用所选的参考体的速度来表示系统的总动量,可以推导出运动学模型。本节将建立一种通用运动学,提出基于总动量的统一表达式的运动学模型。由于先不选择特定的参考体,因此求 FFSR 任一点的速度时不必重新推导出新的参考体所对应的总动量统一表达式,可以减少许多不必要的计算量。

2.2.1 运动学方程

如图 2-3 所示的基本模型由 $n+1$ ($n=6$) 个部分组成,其中 C_0 为卫星本体, C_{1-3} 为右

臂, $C_{4\sim6}$ 为左臂。对于第 i 部分, 动能为:

$$T_i = \frac{1}{2} I_i \cdot \boldsymbol{\omega}_i^2 + \frac{1}{2} m_i \cdot \mathbf{v}_i^2 \quad (2-28)$$

式中, I_i 是第 i 部分相对自身质心 C_i 的转动惯量; $\boldsymbol{\omega}_i$ 是第 i 部分相对自身质心 C_i 的角速度; m_i 是第 i 部分质量; \mathbf{v}_i 是第 i 部分质心的运动速度。

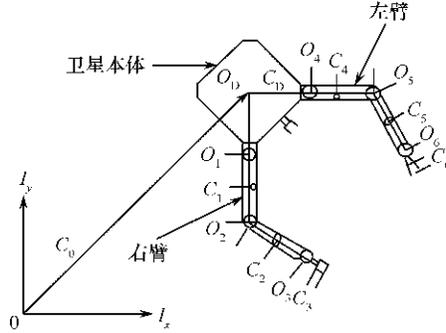


图 2-3 参考航天员的 DFBSR 模型

将上述的动能表达式写成如下矩阵形式:

$$\mathbf{T}_i = \frac{1}{2} I_i \boldsymbol{\omega}_i^2 + \frac{1}{2} m_i \mathbf{v}_i^2 = \frac{1}{2} \boldsymbol{\omega}_i I_i \boldsymbol{\omega}_i + \frac{1}{2} \mathbf{v}_i m_i \mathbf{v}_i = \frac{1}{2} \begin{bmatrix} \boldsymbol{\omega}_i & \mathbf{v}_i \end{bmatrix} \begin{bmatrix} I_i & 0 \\ 0 & m_i \mathbf{1} \end{bmatrix} \begin{bmatrix} \boldsymbol{\omega}_i \\ \mathbf{v}_i \end{bmatrix} \quad (2-29)$$

式中, $[\boldsymbol{\omega}_i \mathbf{v}_i]$ 表示了第 i 部分的运动(包括平动和转动), 记为 t_i ; $\begin{bmatrix} I_i & 0 \\ 0 & m_i \mathbf{1} \end{bmatrix}$ 表示了第 i 部分的惯性矩阵, 记为 M_i ; 0 和 $\mathbf{1}$ 分别为 3×3 零矩阵和单位矩阵。

第 i 部分的相对于自身质心 C_i 的角动量和线动量分别为:

$$L_{ii} = I_i \boldsymbol{\omega}_i = \frac{\partial \mathbf{T}_i}{\partial \boldsymbol{\omega}_i} \quad (2-30)$$

$$K_{ii} = m_i \mathbf{v}_i = \frac{\partial \mathbf{T}_i}{\partial \mathbf{v}_i} \quad (2-31)$$

任意选择机器人上某一部分 P 作为参考体, 机器人第 i 部分相对于 P 部分的线动量和角动量分别为:

$$L_{iP} = \frac{\partial \mathbf{T}_i}{\partial \boldsymbol{\omega}_P} \quad (2-32)$$

$$K_{iP} = \frac{\partial \mathbf{T}_i}{\partial \mathbf{v}_P} \quad (2-33)$$

整个系统相对于 P 部分质心的动量为:

$$L_P = \frac{\partial \mathbf{T}}{\partial \boldsymbol{\omega}_P} = \partial \sum_{i=0}^n \mathbf{T}_i / \partial \boldsymbol{\omega}_P = \sum_{i=0}^n I_i \boldsymbol{\omega}_i \frac{\partial \boldsymbol{\omega}_i}{\partial \boldsymbol{\omega}_P} \quad (2-34)$$

$$K_P = \frac{\partial \mathbf{T}}{\partial \mathbf{v}_P} = \partial \sum_{i=0}^n \mathbf{T}_i / \partial \mathbf{v}_P = \sum_{i=0}^n m_i \mathbf{v}_i \frac{\partial \mathbf{v}_i}{\partial \mathbf{v}_P} \quad (2-35)$$

定义扩展动量为：

$$\mathbf{h} = \begin{bmatrix} L \\ K \end{bmatrix} \quad (2-36)$$

通过变换定律,可得整个系统相对于某一惯性固定点的扩展动量为：

$$\mathbf{h} = \begin{bmatrix} L \\ K \end{bmatrix} = \mathbf{B}_P \cdot \mathbf{h}_P \quad (2-37)$$

式中, \mathbf{B}_P 为变换矩阵; \mathbf{h}_P 为整个系统相对于参考体 P 的扩展动量。把 \mathbf{h}_P 写成矩阵形式为：

$$\mathbf{h}_P = \begin{bmatrix} L_P \\ K_P \end{bmatrix} = \begin{bmatrix} \sum_{i=0}^n I_i \boldsymbol{\omega}_i \frac{\partial \boldsymbol{\omega}_i}{\partial \boldsymbol{\omega}_P} \\ \sum_{i=0}^n m_i \mathbf{v}_i \frac{\partial \mathbf{v}_i}{\partial \mathbf{v}_P} \end{bmatrix} = \begin{bmatrix} \frac{\partial \boldsymbol{\omega}_0}{\partial \boldsymbol{\omega}_P} & \frac{\partial \mathbf{v}_0}{\partial \mathbf{v}_P} & \cdots & \frac{\partial \boldsymbol{\omega}_n}{\partial \boldsymbol{\omega}_P} & \frac{\partial \mathbf{v}_n}{\partial \mathbf{v}_P} \end{bmatrix} \begin{bmatrix} I_0 & & & & \\ & m_0 & & & \\ & & \ddots & & \\ & & & I_n & \\ & & & & m_n \end{bmatrix} \begin{bmatrix} \boldsymbol{\omega}_0 \\ \mathbf{v}_0 \\ \vdots \\ \boldsymbol{\omega}_n \\ \mathbf{v}_n \end{bmatrix} \quad (2-38)$$

将 $[\boldsymbol{\omega}_0 \quad \mathbf{v}_0 \quad \cdots \quad \boldsymbol{\omega}_n \quad \mathbf{v}_n]^T$ 记为 t ; $\text{diag}[I_0, m_0, \cdots, I_n, m_n]$ 记为 M , 则

$$\mathbf{h}_P = \left(\frac{\partial t}{\partial t_P} \right)^T \cdot M \cdot t \quad (2-39)$$

式中, t 表明了系统的运动(包括平动转动),称为广义转动; M 表明了系统的转动惯量和质量等特性。

从式(2-39)得扩展动量为：

$$\mathbf{h} = \mathbf{B}_P \cdot \mathbf{h}_P = \mathbf{B}_P \cdot \left(\frac{\partial t}{\partial t_P} \right)^T \cdot M \cdot t \quad (2-40)$$

式(2-40)就是系统总动量的统一表达式,其中, \mathbf{h} 为系统的总动量; t 代表了系统的运动(它由系统各个关节的线速度和角速度组成); t_P 代表了参考体的运动; M 由系统各部分的质量和转动惯量组成。式(2-40)表明,系统的总动量 \mathbf{h} 表示成了所选的参考体的运动速度 t_P 的函数。利用式(2-40),可以进一步进行系统的运动学分析。

2.2.2 运动学分析

下面首先选末端效应器作为参考体推导出运动学方程式。

将末端效应器作为参考体时,系统的广义转动 t 的表达式为：

$$t = \mathbf{T}_e \cdot t_e + \widehat{\mathbf{T}}_m \cdot \dot{\theta} \quad (2-41)$$

$$\dot{\theta} = [\dot{\theta}_1, \dots, \dot{\theta}_n]^T \quad (2-42)$$

式中, t_e 是末端效应器的运动; 标量 $\dot{\theta}_i (i = 1, \dots, n)$ 是旋转关节的角速度; T_e 和 \hat{T}_m 分别是 $6(n+1) \times 6$ 和 $6(n+1) \times n$ 矩阵。选择末端效应器作为参考体的扩展总动量的表达式可写为:

$$h = B_e(I_e \cdot t_e + \hat{I}_m \dot{\theta}) \quad (2-43)$$

式中, B_e 是 6×6 矩阵; I_e 和 \hat{I}_m 分别是 6×6 和 $6 \times n$ 矩阵, 如下所示:

$$I_e = T_e^T \cdot M \cdot T_e \quad \hat{I}_m = T_e \cdot M \cdot \hat{T}_m \quad (2-44)$$

从式(2-43)可以直接得到表示末端效应器的运动和关节角速度之间关系的运动学模型, 即:

$$t_e = J \cdot \dot{\theta} + H \cdot h \quad (2-45)$$

式中, J 和 H 分别是 $6 \times n$ 雅可比矩阵和与 h 相关的 6×6 矩阵, 如下所示:

$$J = -I_e^{-1} \cdot \hat{I}_m, \quad H = I_e^{-1} \cdot B_e^{-1} \quad (2-46)$$

式中, I_e 是一个正对称阵, 所以 I_e^{-1} 存在。从式(2-46)可以看出, 矩阵 J 的表达式比广义雅可比矩阵(GJM)简单。

2.3 DF SR 的运动特性

2.3.1 姿态干扰特性

由于 FFSR 的基座不固定, 机械手的运动会对本体产生反作用力和力矩, 引起本体的姿态和位置改变。图 2-4 显示了当机器人的左臂以角速度 $(1/\pi)/(\text{rad} \cdot \text{s}^{-1})$ 运动时, 本体姿态的变化过程。图(a)为机器人的初始状态, 图(b)显示了左臂运动过程和本体姿态的变化, 图(c)为本体姿态角随时间变化的情况。

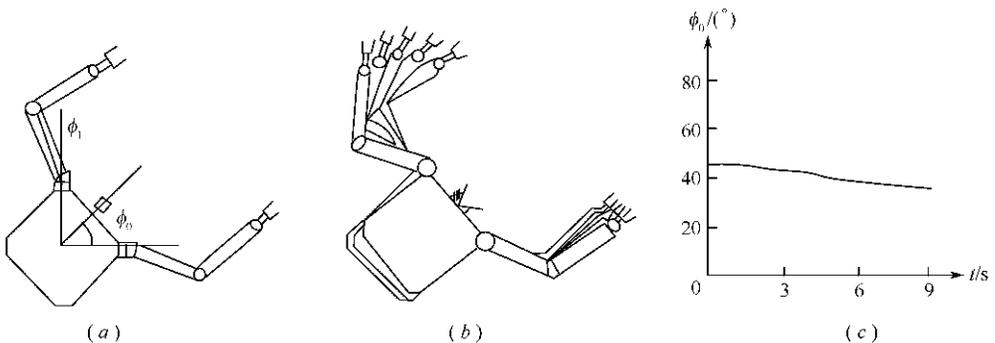


图 2-4 FFSR 的姿态干扰特性

(a) 初始状态; (b) 左臂运动过程; (c) 本体姿态变化。

2.3.2 冗余特性

当图 2-2 所示的机器人在地面工作时,它是一个非冗余机器人。而当它在微重力环境下工作时,它变成一个冗余机器人。这是由于机器人输入变量的维数大于末端效应器变量的维数而形成的。例如对于图 2-5 所示的机器人,它的末端可用一个三维变量描述。由于 FFSR 的基座不固定,因此机器人的输入变量是一个四维变量,从而变成一个冗余机器人。由于冗余性的存在,对于世界空间中的任意一点,机器人可以以无穷多的姿态到达此点。由于冗余性的存在,FFSR 与地面机器人相比具有更高的灵活性,但其控制算法将比地面机器人更复杂。

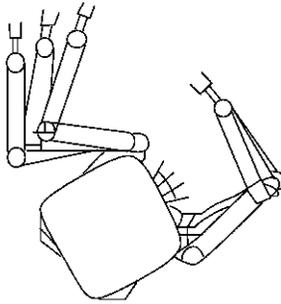


图 2-5 FFSR 的冗余性

2.3.3 非完整性

在空间微重力环境下,FFSR 系统不受外力,因此满足线动量和角动量守恒方程。DFFSR 系统的线动量守恒可写成下式:

$$m_0^1 \dot{\mathbf{r}}_0 + \sum_{i=1}^n m_i^{11} \dot{\mathbf{r}}_i^1 + \sum_{i=1}^m m_i^{r1} \dot{\mathbf{r}}_i^r = 0 \quad (2-47)$$

式(2-47)左边代表 FFSR 系统各连杆的线动量之和。将式(2-47)对时间积分,可将其表示为系统中各连杆在关节坐标系中的位置向量。式(2-47)是可积的,因此是一个完整约束。DFFSR 的角动量守恒方程为:

$${}^1I_0^1 \boldsymbol{\omega}_0 + m_0^1 \mathbf{r}_0 \times {}^1\dot{\mathbf{r}}_0 + \sum_{i=1}^n ({}^1I_i^1 \boldsymbol{\omega}_i^1 + m_i^{11} \mathbf{r}_i^1 \times {}^1\dot{\mathbf{r}}_i^1) + \sum_{i=1}^m ({}^1I_i^r \boldsymbol{\omega}_i^r + m_i^{r1} \mathbf{r}_i^r \times {}^1\dot{\mathbf{r}}_i^r) = 0 \quad (2-48)$$

式(2-48)是不可积的,因此它是一个非完整约束。这说明当 FFSR 按照不同的路径运动到空间中的某一点时,机器人本体的姿态是不同的。因此 FFSR 的机械手在关节空间中做闭合路径运动时,机器人本体的初始姿态和终止姿态是不同的,图 2-6 给出了机器人的运动过程仿真。图(a)为机器人的初始状态,首先右臂关节 1 顺时针转动 60° ,如图(b)所示,然后右臂关节 2 逆时针旋转 60° ,如(c)所示,在图(d)中右臂关节 1 逆时针转 60° ,图(e)中右臂关节 2 顺时针旋转 60° ,图(f)为机器人初始姿态和终止姿态的对比,从图中可以看出机器人的姿态发生改变。图(g)为机器人各关节的运动过程。

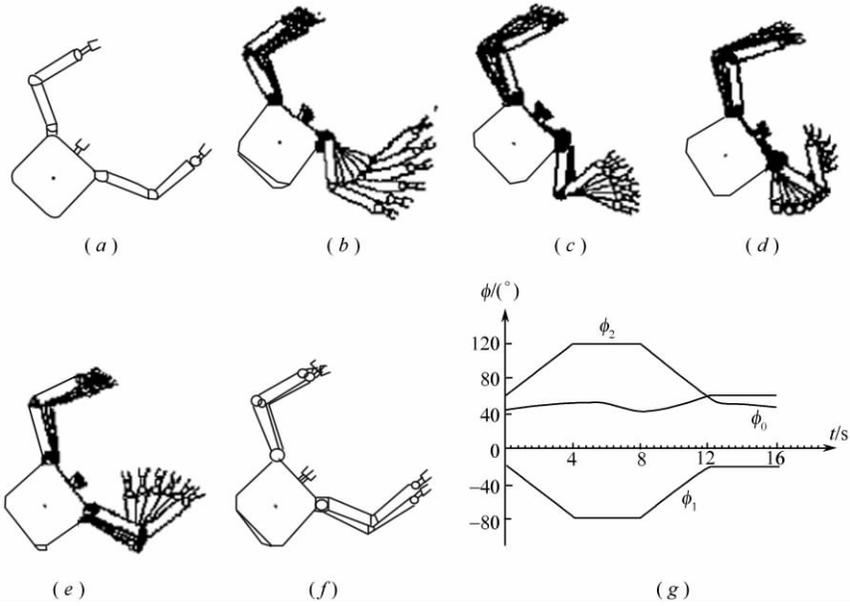


图 2-6 FFSR 的非完整性

(a) 初始状态; (b) 关节 1 运动; (c) 关节 2 运动; (d) 关节 1 运动;
 (e) 关节 2 运动; (f) 初始和终止状态; (g) 运动过程。

2.4 DFFSR 的工作空间及其划分

2.4.1 工作空间

工作空间的概念是 B. Roth 于 1975 年提出的,从几何方面讨论机器人的工作性能。工作空间是指机器人正常运行时,手腕机械接口坐标系的原点能在空间活动的最大范围,或者说该原点可达点占有的空间体积。工作空间也可称为可达空间,通常记作 $W(P)$ 。

当 DFFSR 不采用反作用轮或反作用喷气装置来维持机器人本体的姿态或位置不变时,其工作空间将与地面机器人的工作空间不同。对图 2-2 所示的 6 自由度 DFFSR 来说,它的质量及连杆参数如图 2-7 所示。

假如该机器人的位置和姿态由喷气装置加以限制时,它的工作空间与地面机器人工作空间一致。以左臂为例,它的工作空间为以 P_1^l 为原点,以 R_C 为半径的圆,如图 2-8 所示。它的半径 R_C 为:

$$|R_C| = \left| \sum_{i=1}^3 (a_i^l + b_i^l) \right| \quad (2-49)$$

地面机器人的工作空间称为受限工作空间,在图中用 $CW(P)$ 表示。当机器人的位置和姿态不受控制时,它的工作空间称为自由工作空间,在图中用 $FW(P)$ 表示。它的圆心为系统质心。令 r_G 表示指向系统质心的向量,则

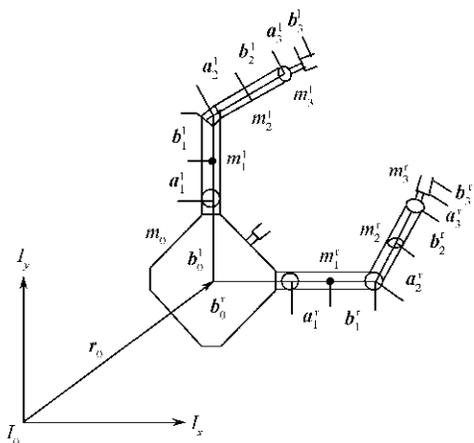


图 2-7 DFBSR 物理参数模型

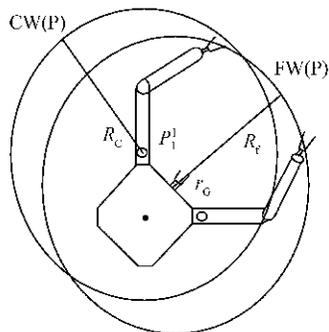


图 2-8 FFSR 与地面机器人的工作空间

$$r_G = r_0 + \frac{\sum_{i=1}^3 \left[\left(\sum_{k=0}^{i-1} b_k^l + \sum_{k=1}^i a_k^l \right) m_i^l \right] + \sum_{i=1}^3 \left[\left(\sum_{k=0}^{i-1} b_k^r + \sum_{k=1}^i a_k^r \right) m_i^r \right]}{W} \quad (2-50)$$

式中, \$W\$ 为系统总质量, 且

$$W = m_0 + \sum_{i=1}^3 m_i^r + \sum_{i=1}^3 m_i^l \quad (2-51)$$

则工作空间 \$FW(P)\$ 的半径为:

$$|R_f| = \left| \sum_{i=1}^4 L_i^r + \sum_{i=1}^4 L_i^l \right| \quad (2-52)$$

式中

$$L_1^r = p_1^r \quad (2-53a)$$

$$L_2^r = p_2^r + q_2^r \quad (2-53b)$$

$$L_3^r = p_3^r + q_3^r \quad (2-53c)$$

$$L_4^r = q_4^r \quad (2-53d)$$

$$L_1^l = p_1^l \quad (2-53e)$$

$$L_2^l = p_2^l + q_2^l \quad (2-53f)$$

$$L_3^l = p_3^l + q_3^l \quad (2-53g)$$

$$L_4^l = q_4^l \quad (2-53h)$$

$$p_i^r = - \frac{b_{i-1}^r \cdot \sum_{k=i}^3 m_k^r}{W} \quad (i = 1, 2, 3) \quad (2-54)$$

$$q_i^r = - \frac{a_{i-1}^r \cdot \sum_{k=i-1}^3 m_k^r}{W} \quad (i = 2, 3, 4) \quad (2-55)$$

$$p_i^l = \frac{b_{i-1}^l (m_0 + \sum_{k=1}^3 m_k^r + \sum_{k=0}^{i-1} m_k^l)}{W} \quad (i = 1, 2, 3) \quad (2-56)$$

$$q_i^l = \frac{a_{i-1}^l (m_0 + \sum_{k=1}^3 m_k^r + \sum_{k=0}^{i-2} m_k^l)}{W} \quad (i = 2, 3, 4) \quad (2-57)$$

2.4.2 工作空间划分

在工作空间内,末端效应器可以以任意姿态到达的点所构成的工作空间称为灵活工作空间,记作 $W^P(P)$ 。工作空间去掉灵活工作空间余下的部分称为附属工作空间,或次工作空间,记作 $W^S(P)$ 。根据定义有:

$$W(P) = W^P(P) + W^S(P) \quad (2-58)$$

灵活工作空间可分作两类: I 类灵活工作空间是末端效应器以所有姿态到达的点所构成空间; II 类灵活工作空间是末端效应器只能以有限个姿态到达的点所构成的空间。它们分别表示为: $W^{P^I}(P)$ 和 $W^{P^{II}}(P)$ 。以图 2-7 所示的机器人为例,它的灵活工作空间为图 2-9 中由圆 C_2 和 C_3 所构成的环形区域。

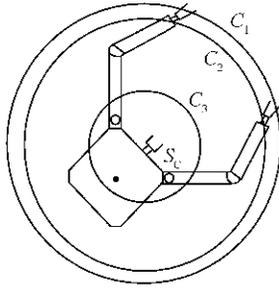


图 2-9 FFSR 的灵活工作空间

在图 2-9 中,圆 C_2 和圆 C_3 的圆心均为系统质心,其中圆 C_2 的半径 R_{C_2} 可由下式计算:

$$|R_{C_2}| = \left| \sum_{i=1}^4 L_i^r + \sum_{i=1}^3 L_i^l - L_4^l \right| \quad (2-59)$$

圆 C_3 的半径为:

$$|R_{C_3}| = \left| \sum_{i=1}^4 L_i^r + \sum_{i=1}^2 L_i^l - L_3^l + L_4^l \right| \quad (2-60)$$

从图 2-9 中可以得出以下结论:①圆 C_3 的内部和圆 C_1 和圆 C_2 所形成的环形区域为

次工作空间 $W^S(P)$ 。② 工作空间的边界,即圆 C_1 的边界为空间机器人的奇异点。当机器人处在奇异点时,将无法对机器人进行控制,因此应避免机器人运行到奇异点。③ FFSR 的最后一个连杆的长度越长,则工作空间 C_1 的半径越大,而相应的灵活工作空间由于 C_3 半径的增大而减小。

2.4.3 连杆质量对工作空间的影响

与地面机器人所不同的是,当 FFSR 各连杆的质量变化时,它的工作空间也随之变化。而地面机器人的工作空间与质量无关。机器人本体的质量对工作空间的大小的影响如图 2-10 所示。

机器人的末端效应器捕捉目标以后,最后一个连杆的质量发生变化。这时它的工作空间将随之变化。目标质量对工作空间大小的影响如图 2-11 所示。

从图 2-10 和图 2-11 中,可以得出如下结论:① 当机器人本体质量增大时,工作空间也随之增大。② 当目标质量增加时,机器人的工作空间将减小,特别是当目标质量变得足够大时,工作空间的半径将减小为零。

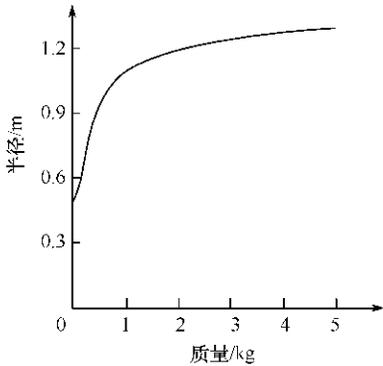


图 2-10 本体质量对工作空间的影响

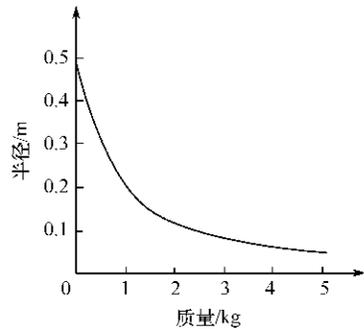


图 2-11 目标质量对工作空间的影响

2.5 多臂 FFSR 运动学模型

本节首先给出描述多臂 FFSR 机械臂末端运动速度和机器人各关节运动角速度关系的广义雅可比矩阵和对本体的位置和姿态干扰方程,分析 FFSR 的运动特性及其局限性,讨论 FFSR 分解运动速度控制和分解运动加速度控制方法。

2.5.1 运动学模型

考虑如图 2-12 所示的树形结构多臂 FFSR 系统,由作为机器人本体的卫星和搭载其上的旋转型关节的刚性机械臂组成。整个系统在惯性空间中自由飞行或浮游,没有反作用喷气装置和姿态控制设备运行,内力仅由关节电机产生。因此,整个系统满足线动量和角动量守恒。

设 FFSR 由编号从 1 到 i 的 i 个机械臂组成,本体编号为 0。定义第 k 个机械臂的连杆

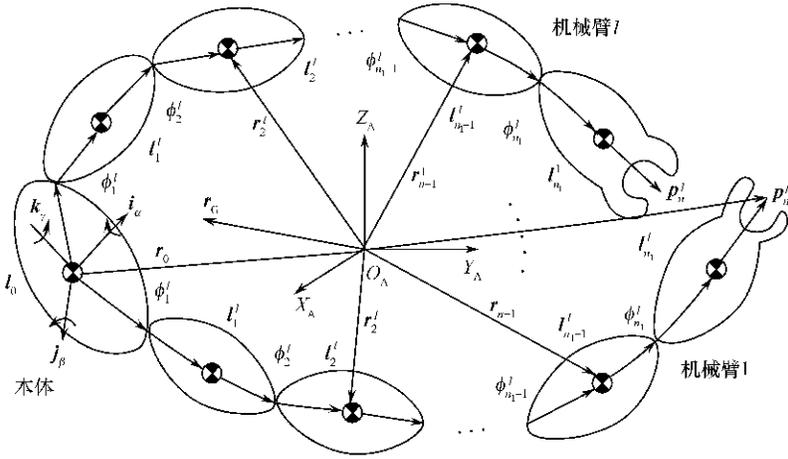


图 2-12 多臂 FFSR 模型

记为 1 到 n_k , 每一个臂的第 i 个关节为第 i 连杆和第 $i-1$ 连杆间的关节, 第 k 个机械臂的第 j 个关节角为 ϕ_j^k , 右上标 k 表示第 k 个机械臂, 右下标 j 表示该机械臂的第 j 个关节角或连杆。本体姿态角表示为 ϕ_s 。图中及式推导中的符号定义如下:

定义关节向量空间记为: $\phi_m = (\phi_1^1, \dots, \phi_{n_1}^1, \phi_1^2, \dots, \phi_{n_2}^2, \dots, \phi_1^l, \dots, \phi_{n_l}^l)^T$;

系统总的自由度则为: $n = \sum_{k=1}^{k=l} n_k$;

$r_0 \in R^3$: 机器人本体质心位置向量;

m_0 : 机器人本体的质量;

$I_0 \in R^{3 \times 3}$: 机器人本体相对于其质心的惯性矩阵;

ω_0 : 机器人本体姿态变化的角速度;

$r_g \in R^3$: 整个机器人系统质心的位置向量;

W : 整个机器人系统质量;

$r_i^k \in R^3$: 第 k 个机械臂的第 i 个连杆的质心位置向量;

m_i^k : 第 k 个机械臂的第 i 个连杆的质量;

$I_i^k \in R^{3 \times 3}$: 第 k 个机械臂的第 i 个连杆相对于其质心的惯性矩阵;

ω_i^k : 第 k 个机械臂的第 i 个关节的角速度;

$P_j^k \in R^3$: 第 k 个机械臂的第 j 个关节的位置矢量;

$P_n^k \in R^3$: 第 k 个机械臂末端效应器的位置向量;

$K_j^k \in R^3$: 第 k 个机械臂的第 j 个关节旋转坐标的单位矢量;

$E \in R^{3 \times 3}$: 单位矩阵。

其中未注明的所有变量均相对于惯性坐标系, 惯性坐标系的原点一般置于系统质心。

记向量 $r = (x, y, z)^T$, 引入向量算子 \tilde{r} 如下:

$$\tilde{\mathbf{r}} = \begin{bmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{bmatrix}$$

2.5.2 运动学方程

FFSR 第 k 个机械臂的末端线速度和角速度分别记为 $\dot{\mathbf{P}}_n^k$ 和 $\boldsymbol{\omega}_n^k$, 根据 FFSR 的几何结构和连杆间的运动关系可得:

$$\dot{\mathbf{P}}_n^k = \mathbf{r}_0 + \boldsymbol{\omega}_0 \times (\mathbf{P}_n^k - \mathbf{r}_0) + \sum_{i=1}^{n_k} [\mathbf{K}_i^k \times (\mathbf{P}_n^k - \mathbf{P}_i^k)] \cdot \dot{\boldsymbol{\phi}}_i^k \quad (2-61)$$

$$\boldsymbol{\omega}_n^k = \boldsymbol{\omega}_0 + \sum_{i=1}^{n_k} \mathbf{K}_i^k \dot{\boldsymbol{\phi}}_i^k \quad (2-62)$$

式(2-61) 和式(2-62) 可以写成表达式:

$$\begin{bmatrix} \dot{\mathbf{P}}_n^k \\ \boldsymbol{\omega}_n^k \end{bmatrix} = \begin{bmatrix} \mathbf{E} & -\tilde{\mathbf{P}}_{0n}^k \\ 0 & \mathbf{E} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{r}}_0 \\ \boldsymbol{\omega}_0 \end{bmatrix} + \begin{bmatrix} \mathbf{J}_{mp}^k \\ \mathbf{J}_{m\omega}^k \end{bmatrix} \cdot \dot{\boldsymbol{\phi}}_m \quad (2-63)$$

式中

$$\tilde{\mathbf{P}}_{0n}^k = \mathbf{P}_n^k - \mathbf{r}_0^k$$

$$\mathbf{J}_{mp}^k = [0 \cdots 0 \quad \mathbf{K}_1^k \times (\mathbf{P}_n^k - \mathbf{P}_1^k) \quad \mathbf{K}_2^k \times (\mathbf{P}_n^k - \mathbf{P}_2^k) \quad \cdots \quad \mathbf{K}_{n_k}^k \times (\mathbf{P}_n^k - \mathbf{P}_{n_k}^k) \quad 0 \cdots 0]$$

$$\mathbf{J}_{m\omega}^k = [0 \cdots 0 \quad \mathbf{K}_1^k \quad \mathbf{K}_2^k \quad \cdots \quad \mathbf{K}_{n_k}^k \quad 0 \quad \cdots \quad 0]$$

记第 k 个机械臂的末端的任务向量为 $\mathbf{v}^k = (\dot{\mathbf{P}}_n^k, \boldsymbol{\omega}_n^k)^T$, 显然它和 FFSR 本体的线速度 $\dot{\mathbf{r}}_0$ 和角速度 $\boldsymbol{\omega}_0$ 有关, 而 $\dot{\mathbf{r}}_0, \boldsymbol{\omega}_0$ 受机械臂运动的干扰。但考虑到其满足动量守恒定律, 则根据 FFSR 系统的线动量和角动量守恒有:

$$\mathbf{P} = m_0 \dot{\mathbf{r}}_0 + \sum_{k=1}^l \sum_{i=1}^{n_k} m_i^k \dot{\mathbf{r}}_i^k \quad (2-64)$$

$$\mathbf{L} = I_0 \boldsymbol{\omega}_0 + m_0 \mathbf{r}_0 \times \dot{\mathbf{r}}_0 + \sum_{k=1}^l \sum_{i=1}^{n_k} (I_i^k \boldsymbol{\omega}_i^k + m_i^k \mathbf{r}_i^k \times \dot{\mathbf{r}}_i^k) \quad (2-65)$$

式(2-64) 和式(2-65) 写成 $\dot{\mathbf{r}}_0, \boldsymbol{\omega}_0, \dot{\boldsymbol{\phi}}_m$ 的表达式:

$$\begin{bmatrix} \mathbf{P} \\ \mathbf{L} \end{bmatrix} = \begin{bmatrix} \mathbf{W}\mathbf{E} & -\mathbf{W}\tilde{\mathbf{r}}_{0g} \\ \mathbf{W}\tilde{\mathbf{r}}_g & I_w \end{bmatrix} \begin{bmatrix} \dot{\mathbf{r}}_0 \\ \boldsymbol{\omega}_0 \end{bmatrix} + \begin{bmatrix} \mathbf{J}_{T\omega} \\ \mathbf{I}_\phi \end{bmatrix} \cdot \dot{\boldsymbol{\phi}}_m \quad (2-66)$$

式中

$$\begin{aligned} \mathbf{J}_{T\omega} &= \sum_{k=1}^l \sum_{i=1}^{n_k} m_i^k \mathbf{J}_{Ti}^k \\ \mathbf{J}_{Ti}^k &= [0, \dots, 0, \mathbf{K}_1^k \times (\mathbf{r}_i^k - \mathbf{P}_1^k), \dots, \mathbf{K}_i^k \times (\mathbf{r}_i^k - \mathbf{P}_i^k), 0, \dots, 0] \\ \mathbf{I}_\omega &= \sum_{k=1}^l \sum_{i=1}^{n_k} (\mathbf{I}_i^k - m_i^k \tilde{\mathbf{r}}_i^k \tilde{\mathbf{r}}_{0i}^k) + \mathbf{I}_0 \\ \mathbf{I}_\phi &= \sum_{k=1}^l \sum_{i=1}^{n_k} (\mathbf{I}_i^k \mathbf{J}_{Ri}^k + m_i^k \tilde{\mathbf{r}}_i^k \mathbf{J}_{Ti}^k) \\ \mathbf{J}_{Ri}^k &= [0, \dots, 0, K_1^k, \dots, K_i^k, 0, \dots, 0] \\ \tilde{\mathbf{r}}_{0g}^k &= \mathbf{r}_g - \mathbf{r}_0, \tilde{\mathbf{r}}_{0i}^k = \tilde{\mathbf{r}}_i^k - \mathbf{r}_0 \end{aligned}$$

不失一般性,设初始状态时系统总的线动量和角动量为零,则由式(2-66)求解方程组可得:

$$\mathbf{I}_S \boldsymbol{\omega}_0 + \mathbf{I}_m \dot{\boldsymbol{\phi}}_m = 0 \quad (2-67)$$

$$\dot{\mathbf{r}}_0 = -(\mathbf{J}_{T\omega}/W + \tilde{\mathbf{r}}_{0g} \mathbf{I}_S^{-1} \mathbf{I}_m) \cdot \dot{\boldsymbol{\phi}}_m = \mathbf{J}_v \dot{\boldsymbol{\phi}}_m \quad (2-68)$$

式中

$$\mathbf{I}_S = \mathbf{I}_\omega + W \tilde{\mathbf{r}}_g \tilde{\mathbf{r}}_{0g}$$

$$\mathbf{I}_m = \mathbf{I}_\phi - \tilde{\mathbf{r}}_g \mathbf{J}_{T\omega}$$

$\mathbf{I}_S \in R^{3 \times 3}$ 和 $\mathbf{I}_m \in R^{3 \times n}$ 分别定义为 FFSR 本体和手臂的广义惯性矩阵。式(2-67)和式(2-68)描述的便是多臂 FFSR 机械臂运动时对机器人本体姿态和位置的干扰方程,将式(2-68)代入式(2-63)可得:

$$\mathbf{v}^k = \begin{bmatrix} \dot{\mathbf{P}}_n^k \\ \dot{\boldsymbol{\omega}}_n^k \end{bmatrix} = \begin{bmatrix} -\tilde{\mathbf{P}}_{0n}^k \\ \mathbf{E} \end{bmatrix} \cdot \dot{\boldsymbol{\phi}}_S + \begin{bmatrix} \mathbf{J}_v + \mathbf{J}_{mp}^k \\ \mathbf{J}_{m\omega}^k \end{bmatrix} \cdot \dot{\boldsymbol{\phi}}_m = \mathbf{J}_S^k \dot{\boldsymbol{\phi}}_S + \mathbf{J}_m^k \dot{\boldsymbol{\phi}}_m \quad (2-69)$$

由式(2-67)求得:

$$\boldsymbol{\omega}_0 = -\mathbf{I}_S^{-1} \mathbf{I}_m \dot{\boldsymbol{\phi}}_m = \mathbf{J}_\omega \dot{\boldsymbol{\phi}}_m \quad (2-70)$$

代入式(2-69)可得:

$$\mathbf{v}^k = (\mathbf{J}_m^k - \mathbf{J}_S^k \mathbf{I}_S^{-1} \mathbf{I}_m) \cdot \dot{\boldsymbol{\phi}}_m = \mathbf{J}^{\cdot k} \cdot \dot{\boldsymbol{\phi}}_m \quad (2-71)$$

式中

$$\mathbf{J}^{\cdot k} = \mathbf{J}_m^k - \mathbf{J}_S^k \mathbf{I}_S^{-1} \mathbf{I}_m$$

$\mathbf{J}^{\cdot k}$ 即是多臂 FFSR 第 k 个机械臂末端运动速度的广义雅可比矩阵,它反应了机器人

本体自由浮游情况下,第 k 个机械臂末端执行器的线速度和角速度同 FFSR 关节角速度间的关系。

2.6 小 结

本章主要研究 FFSR 运动学模型及运动特性。首先,分析了双臂 FFSR 的运动模型,给出双臂 FFSR 的广义雅可比矩阵;同时研究了双臂 FFSR 的通用运动学模型。其次,给出了与运动规划密不可分的不同于地面机器人的三个特性:姿态干扰特性、非完整性及冗余特性。第三,研究了双臂 FFSR 的工作空间及其划分问题。最后,研究了多臂 FFSR 的运动模型,给出了多臂 FFSR 的在空间微重力环境下的广义雅可比矩阵。

第三章 基于神经网络的空间机器人参数辨识

广义雅可比矩阵(GJM)不仅和FFSR的几何结构有关,还与其各部分的惯性参数如质量、转动惯量等有关。但是FFSR本体的惯性参数是随着载荷变化及燃料的消耗而发生变化,因此必须对其进行在线实时辨识。考虑到人工神经网络具有很强的学习记忆能力,提出用前馈神经网络学习FFSR的姿态干扰特性从而辨识其惯性参数的方法。为此首先分析神经网络的逼近能力和常用的学习算法,注意到BP算法收敛速度慢、易陷入局部极小点,提出一种非线性同伦综合学习算法,在增强避免陷入局部极小能力的同时,又较大地提高了收敛速度。最后将该算法用于FFSR的参数辨识,研究具体的实现过程和应注意的问题,并针对实验模型进行计算机仿真。

3.1 前馈神经网络的逼近能力和学习算法

多层前馈神经网络是目前最常用的神经网络模型,它的逼近能力和训练算法是其应用的关键。

3.1.1 多层前馈神经网络的逼近能力

多层前馈神经网络的输入输出关系,可以看成是一种映射关系,即每一种输入对应一组输出。由于神经元作用函数的非线性,网络实现的是复杂的非线性映射,具有函数逼近和大规模并行分布处理能力。关于这类网络对非线性的逼近能力,Hornik等人进行了很多研究,此处直接给出结论。

定理 3.1 对任意给定的函数 $f(x_1, x_2, \dots, x_n)$ 和误差精度 $\epsilon > 0$, 存在一个三层前馈神经网络, 其隐层神经元的作用函数 $\sigma(x)$ 具有阈值特性, 输入、输出层为线性神经元。则总的输入输出关系: $Y = \hat{f}(x_1, x_2, \dots, x_n)$ 能以规定的精度逼近给定的函数 $f(x_1, x_2, \dots, x_n)$ 。

另外, Hornik 等人的研究还表明, 三层前馈神经网络不仅能以任意的精度逼近任意函数, 还能以任意精度逼近其各阶导数。本定理为用神经网络学习 FFSR 的姿态干扰特性, 辨识其惯性参数奠定了理论基础。

3.1.2 多层前馈神经网络的学习算法

目前主要有: ① 误差反传(BP) 算法及其各种改进算法; ② 其他实现全局最优的训练算法, 如模拟退火算法、趋药分类法、遗传算法等基于随机优化技术的全局训练算法、基于线性最小二乘技术的训练算法和一些优化方法, 如卡尔曼滤波、同伦优化等, 均收到较好的效果, 但仍有许多实际应用问题需要解决。

BP 算法理论因其依据坚实,推导过程严谨,所得公式对称优美,物理概念清楚,通用性强而得到了广泛的应用。但针对与其不足之处如收敛速度慢、易限于局部极小及所得的网络性能较差等,许多学者提出了很多改进算法。BP 算法中神经元的输入输出关系的通用方程为:

$$\begin{aligned} x_i(l+1) &= \sum_{j=0}^{N_l} \omega_{ij}(l+1)y_j(l) \\ y_i(l+1) &= g_i(x_i(l+1)) \end{aligned} \quad (3-1)$$

式中, $1 \leq i \leq N_{l+1}$ 表示第 $l+1$ 层的第 i 个节点; $1 \leq l \leq L-1$ 表示层数; g_i 为非减可微函数; x_i 和 y_i 分别为第 i 个神经元的输入和输出; $y_i(0)$ 代表网络的输入; $\omega_{ij}(l)$ 表示第 l 层第 i 个神经元的阈值。

BP 算法采用梯度下降技术使希望的目标和真实值之间的误差平方和最小,具体公式如下:

$$\Delta\omega_{ij}(l,k) = -\eta \delta_i(l,k) g'_i(x_i(l,k)) y_j(l-1,k) + \alpha \Delta\omega_{ij}(l,k-1) \quad (3-2)$$

$$\delta_i(l,k) = \begin{cases} d_i^{(p)} - y_i^{(p)}(L) & l = L \\ \sum_{j=1}^{N_{l+1}} \delta_j(l+1,k) \cdot g'_j(x_j(l+1,k)) \omega_{ji}(l+1,k) & 1 \leq l < L \end{cases} \quad (3-3)$$

式中, $d_i^{(p)}$ 为教师信号; η 是学习率或迭代步长; α 是动量因子。

目前主要有以下几种改进算法:①改进误差函数法。②改进激发函数方法。③变学习率 η 方法。④变动量因子 α 方法。⑤以上几种方法的综合算法。这些方法在提高收敛速度和避免陷入局部极小方面,取得了一定的效果,但仍未得到很满意的答案,许多工作需要进一步研究。

3.2 非线性同伦综合学习算法

3.2.1 同伦 BP 学习算法

由于 BP 算法的求解空间是极其复杂的多维曲面,存在许多局部极小点,常规 BP 算法是沿能量函数曲面的负梯度方向下行的以及随机设置初始权值,更增加了陷于局部极小的可能性。考虑到 BP 算法是求极小值问题,可以转化为求能量函数的导数的零点问题。因此,对能量函数极值的跟踪可以转化为对其梯度函数零点的跟踪,而同伦方法是一种解决非线性问题的较好的方法,为此,许多文献对其在 BP 算法中的应用进行了研究。考虑到高小榕提出的算法具有结构明晰、实现方便的特点,本节以该算法为例来介绍。

先建立一个同伦函数

$$H(t,x) = (1-t) \cdot f_0(x) + t \cdot f_1(x) \quad (3-4)$$

在学习过程中,同伦参数 t 由初始的 $t = 0$ 逐渐变化到终值 $t = 1$ 。神经网络即从对 $f_0(x)$ 的学习过渡到对给定的教师信号 $f_1(x)$ 的学习。每一步仍是通过 BP 算法来达到极小,只

是权重的初值采用上一步的终值。作者给出了三种同伦方法：

$$\text{教师同伦} \quad T(t) = (1-t) \cdot T_g + t \cdot T_f \quad (3-5)$$

$$\text{输入同伦} \quad I(t) = (1-t) \cdot I_g + t \cdot I_f \quad (3-6)$$

$$\text{结构同伦} \quad S(t) = (1-t) \cdot S_g + t \cdot S_f \quad (3-7)$$

式中, T 和 I 是由训练样本向量组成的矩阵, S 视具体问题而定。下标 f 和 g 分别代表终止和初始情况。作者详细讨论了教师同伦和输入同伦算法的实现以及初始训练集的确定。并给出了仿真结果,证明了其在避免陷入局部极小的能力和收敛速度方面明显优于 BP 算法,体现了人类认识事物时所遵循的由简到繁、循序渐进的原则。

3.2.2 非线性同伦 BP 学习算法

考虑到同伦论中采用零点路径跟踪算法时,不同的同伦路径对计算速度的影响,对此进行深入研究。

1. 非线性同伦 BP 学习算法

首先根据同伦函数的定义,本章给出如下同伦函数:

定义 3-1 设 x, y 是 R^k 上的非空子集, k 是正整数。 f_0 和 f_1 都是从 x 到 y 的映射, m, n 为给定的正实数,如果对任意的 $(t, x) \in [0, 1] \times x$ 成立:

$$H(t, x) = (1-t^m) \cdot f_0(x) + t^n \cdot f_1(x) \quad (3-8)$$

则称映射 $H(t, x)$ 是 $f_0(x), f_1(x)$ 间的一个非线性同伦。

针对有教师学习的多层前馈神经网络,记神经网络的期望输出(给定的教师信号)为 $D(D_1, D_2, \dots, D_{N_L})$,根据需要选取适当的初始教师信号 $d(d_1, d_2, \dots, d_{N_L})$,代入式(3-8)中构造其同伦函数可得:

$$T(t) = (1-t^m) \cdot d + t^n \cdot D \quad (3-9)$$

记 $T(t)$ 为同伦教师信号。针对每一个给定的 t ,将 $T(t)$ 作为神经网络的教师信号,应用常规 BP 算法进行网络训练。在学习过程中,同伦参数 t 由初始的 $t = 0$ 逐渐变化到终值 $t = 1$ 。当 $t = 0$ 时, $T(0) = d$,由于 d 可由自己适当选取,因此其全局极小容易求得,下文将给出 d 的一个求法。当 $t = 1$ 时, $T(1) = D$,神经网络即从对 d 的学习过渡到对给定的教师信号 D 的学习。显然,随着 m, n 的取值不同,教师信号从 d 到 D 的过渡路径亦不相同,所以 m, n 的不同取值,亦影响应用该方法进行网络训练速度的快慢。当 $m = n = 1$ 时,即为线性同伦。综上所述,具体的非线性同伦 BP 算法的描述如下。

算法 3-1 非线性同伦 BP 算法

步骤 1 网络训练初始化。

- (1) 将同伦参数均匀分成 N 个离散值,置 $t = t_0$;
- (2) 给出 BP 算法的训练参数 η, α 、网络全局误差 ϵ 以及用于同伦函数的非线性参数 m, n ;
- (3) 初始化神经网络的连接权值和阈值;
- (4) 根据训练样本输入,完成神经网络的前向运算,其输出值作为式(3-9)中的初始

教师信号 d , 显然, 此时的教师信号等于网络的实际输出, $E = 0 < \epsilon$ 成立, 且为全局极小, 置 $t = t_1$ 。

步骤 2 第 i 段学习算法。

(1) 该段初始权值和阈值取为上一段网络收敛时的终值, 同时取前一段的网络输出作为本段同伦算法中的初始教师信号 d , 该段网络收敛后的权值反映的是本段能量函数的极小值;

(2) 将 $d, t = t_i$ 代入式(3-9) 计算新的同伦教师信号 $T(t)$;

(3) 利用 BP 算法进行训练, 如收敛则转步骤 3, 发散则终止。

步骤 3 如果 $i < N - 1$, 则 $i = i + 1$, 回到步骤 2。否则, 算法结束。此时网络权值即为所求。

上述算法中, 由于每一段都应用常规 BP 算法进行网络训练, 因此, 每一步迭代的运算量都是相同的, 即不增加计算的复杂性。另外, 由于每一段的教师信号变化很小, 因此不易陷入局部极小, 并且收敛很快。

2. 算法性能分析

同伦方法的实质是把沿能量函数曲面下行的学习方式改成先找到某一简单问题的全局极小, 然后逐步修改能量函数曲面, 同时跟踪曲面极小点的轨迹以达到待求能量函数的全局极小。从定义 3-1 中可以看出, m, n 的取值决定了同伦路径的形状, 如图 3-1 所示, 也即决定了上述算法不同的迭代过程。

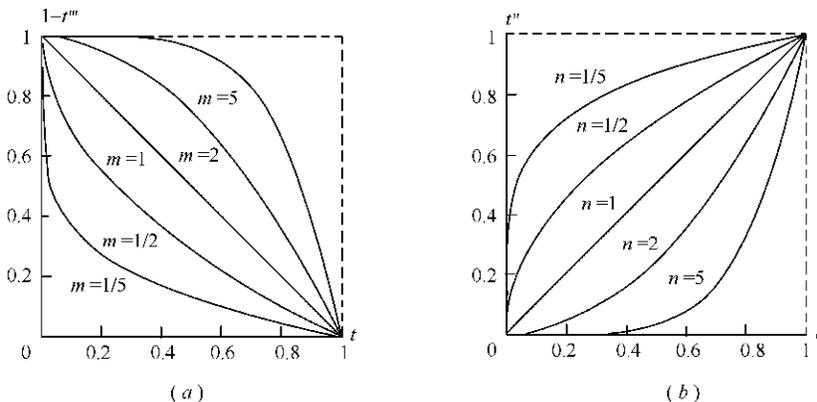


图 3-1 m 和 n 取不同值时的路径变化曲线

同伦算法的核心思想是从给定的初始教师信号 d 开始, 用常规的 BP 算法训练后, 沿着同伦路径逐步改变同伦教师信号 T , 直到完成给定的教师信号 D 的网络训练。在进行神经网络训练时, 同伦参数 t 从 0 向 1 过渡, 使得初始教师信号 d 在同伦教师信号 T 中所占的权重从 1 逐渐向 0 过渡, 与此同时, 给定的教师信号 D 所占的权重则从 0 逐渐向 1 过渡。因此在初始训练时, 给定的教师信号 D 在同伦教师信号 T 中所占权重较小, 初始教师信号 d 在同伦教师信号 T 中所占权重较大。而初始训练时是从 $t = t_0 = 0$ 时的初始教师信号 d 的全局极小开始, 沿着非线性同伦路径进行过渡。因此, 在初始训练时, 沿着非线性同伦路径进行过渡时可以变化得快一些, 到该神经网络经过初步学习训练后, 初始教师信号 d 和给定的教师信号 D 在同伦教师信号 T 中所占的权重有了较显著的对比变化后, 教师信

号 D 所占的比重越来越大,此时应该适当减慢同伦路径变化的过程,以利于神经网络的收敛。这样就可以获得比同伦路径线性变化时更快的神经网络收敛速度。从图 3-1 可知,当 t 从 0 向 1 刚刚开始变化,且 $m < 1$ 时, $1 - t^m$ 比 $1 - t$ 变化得要快,而当 $n < 1$ 时, t^n 比 t 变化得要快, t 过渡一段时间后(和 $m、n$ 的具体取值有关),则正好相反。因此当 $m < 1, n < 1$ 时,满足上述理论分析的条件,此时神经网络训练应该比 $m = n = 1$ 时具有更快的收敛速度;相反,当 $m > 1, n > 1$ 时,神经网络训练应该比 $m = n = 1$ 时的收敛速度要慢。

由于在理论上无法事先求得用 BP 算法训练神经网络收敛时的迭代步数,因此本章进行了大量的仿真,证明了所提算法对训练性能提高的有效性。

3.2.3 非线性同伦综合学习算法

根据同伦 BP 算法的理论基础,其主要的优点是避免陷入局部极小点的能力很强,尽管在每一段中教师信号变化比较小,考虑到 BP 算法的收敛速度比较慢,而该方法的迭代次数为各段迭代次数的总和,因此同伦 BP 算法在提高收敛速度方面的能力可以进一步提高。从上述算法可以看出,在每一段的训练过程中均采用常规 BP 算法,即该算法不增加每一步的计算量及计算复杂性,其避免陷入局部极小的能力均来自同伦函数。因此,可以认为,如果将同伦算法中的 BP 算法用其他改进的快速神经网络训练算法替换,则可以在提高神经网络训练速度的同时,达到提高其避免陷入局部极小的能力,从而在收敛性和收敛速度方面得到更佳的网络训练效果,本章称之为非线性同伦综合学习算法。此处仅以 Parlos 提出的快速算法为例,做一具体说明。用 Parlos 算法代替同伦算法中的 BP 算法,即可生成非线性同伦 Parlos 算法,其在收敛速度方面比同伦 BP 算法快,避免陷入局部极小的能力方面要比 Parlos 算法好,它综合了 Parlos 算法收敛速度快和非线性同伦算法避免陷入局部极小点能力强的这两种算法的优点。

3.2.4 各种算法的性能比较

为了比较所提出的算法的优越性,此处给出了几个用于仿真的实例。进行参数辨识时,还将进一步进行比较。考虑到一些文献中详细讨论了线性同伦和常规 BP 算法的比较和分析,得出了此方法在收敛速度和避免陷入局部极小的能力方面均明显优于常规 BP 算法的结论。在此仅讨论 $m、n$ 取不同参数时与线性同伦的比较。因此性能比较可分为以下两个内容。

1. $m、n$ 取不同参数时的非线性同伦与高小榕的线性同伦算法比较

用于“异或(XOR)”问题。网络结构取为,每一阶段的训练均采用 BP 算法调整权值,步长取为 2,动量项系数为 0.8。误差为 0.001。网络初值在 $[-0.5, 0.5]$ 区间随机给定。各种方法每次运行时均采用相同的初值,独立运行 100 次。结果如表 3-1 所示。

表 3-1 不同的参数时非线性同伦 BP 算法的性能比较

仿真内容	$m = n = 1/5$	$m = n = 1/2$	$m = n = 1$	$m = n = 2$	$m = n = 5$
收敛时迭代次数均值	514	598	981	1305	1256

从表 3-1 的仿真结果可以看出, $m、n$ 的不同取值对收敛速度有较大的影响。如 $m = n = 2$ 时比 $m = n = 1$ 时的线性同伦算法收敛速度慢,而 $m = n = 0.5$ 时比 $m = n = 1$ 时

的线性同伦算法收敛速度快,在 $0.5 \sim 0.2$ 范围变化不大,但是当 $m, n < 0.2$ 时,收敛速度提高便很有限,而避免陷入局部极小的能力却有大的下降,因此,建议同伦参数取 $0.2 \sim 0.5$ 之间比较好。由此可见,仿真结果与理论分析是一致的。本章做了大量的仿真,得到相似的结果。关于不同的路径和收敛速度及避免陷入局部极小的能力的关系值得进一步深入研究。

2. 非线性同伦 BP 算法与非线性同伦综合算法的性能比较

非线性同伦综合算法是用 Parlos 提出的快速算法,代替同伦 BP 算法中的 BP 算法形成的,即可生成非线性同伦 Parlos 算法。当然此处的 Parlos 算法可以用其他的神经网络快速算法代替,从而形成新的非线性同伦综合算法以提高整体的训练性能。

用前馈神经网络逼近某一已知函数 $z = f(x, y) = \sin(x)\sin(y)$ 。其中 $x, y \in [-\pi, \pi]$, 在输入空间中均匀选取 121 个样本,其输出值按已知函数计算求出,从而构成样本集。网络结构取为,按文献优化参数 η ,进行 20 次计算机仿真,结果如表 3-2 所示。从结果看出,该算法达到了预期的效果。

表 3-2 不同算法的训练结果比较

学习算法	BP 算法	同伦 BP ($m = n = 0.5$)	Parlos 算法	同伦 Parlos ($m = n = 0.5$)
陷入局部极小次数	5	1	2	0
收敛时迭代次数均值	53217	15492	22036	3541

3.3 FFSR 参数辨识

目前多数 FFSR 的运动规划和运动控制都需要计算广义雅可比矩阵,如捕捉目标的路径规划及其分解运动速度控制等。但是广义雅可比矩阵不但包含了运动学参数还包含了动力学参数,既和 FFSR 的几何结构有关,又与其惯性参数如各部分的质量和转动惯量有关。而许多情况下,一些参数是很难获得的。如机器人本体的惯性参数是随着载荷的变化及燃料的消耗而发生变化,而目标的惯性参数往往是未知的。因此应用 GJM 矩阵进行工作空间计算、路径规划、捕捉目标的运动控制及对本体的反作用力进行预测时,必须进行参数估计及在线辨识。

3.3.1 基于神经网络的参数辨识算法

考虑到 FFSR 可在空间自由飞行,本体的惯性参数是不断变化的,必须进行辨识才能完成姿态干扰最小的运动规划。考虑到惯性参数的变化影响到机械臂运动对本体干扰的大小,因此通过测量手臂运动对本体的干扰就可以求出惯性参数的变化。FFSR 为了满足姿态控制及导航的需要,必然携带姿态敏感器。当机器人处于某一关节空间 ϕ_m 时,驱动手臂以给定的 $\dot{\phi}_m$ 运动,姿态敏感器可测得此时本体姿态角速度 ω_0 和本体平移的线速度 \dot{r}_0 , 可得:

$$J_{\omega} = I_S^{-1} I_m = -\omega_0 \cdot \dot{\phi}_m^+ \quad (3-10)$$

$$\dot{r}_0 = J_v \dot{\phi}_m \quad (3-11)$$

式中, $[\cdot]^+$ 是 $[\cdot]$ 的伪逆矩阵; J_ω 和 J_v 是关节空间向量 ϕ_m 、机器人的几何参数和各部分惯性参数的函数。由于机械臂各连杆的几何参数和惯性参数是不变的, 式(3-10) 和式(3-11) 即是关于本体惯性参数或目标惯性参数的方程。Y. Murotsu 和 S. Tsujio, Y. Murotsu 和 K. Senda 分别提出了基于该原理的参数辨识方法, 即首先利用线动量和角动量守恒定律生成关于未知参数的方程组, 然后在驱动机械臂运动的同时, 在线测量其对本体姿态和位置干扰的角速度和线速度, 进而求解方程组的参数辨识算法。本书已在绪论中分析了这些方法的优缺点, 因此期望寻求新的解决办法。

由于惯性参数的变化影响对本体干扰的大小, 如果用神经网络学习在本体惯性参数变化的情况下, 空间机器人的干扰特性, 则通过测量手臂运动对本体的干扰就可以利用神经网络求出未知的惯性参数。根据定理 3-1 可知, 多层前馈神经网络可以任意精度逼近任何连续函数, 因此可以任意精度逼近式(3-10) 所描述的姿态干扰特性。考虑到本体的线速度在空中比较难于测量, 因此不学习式(3-11) 所描述的本体位置干扰特性。这就是该辨识算法的理论基础。其神经网络的结构如图 3-2 所示, 具体的算法如下:

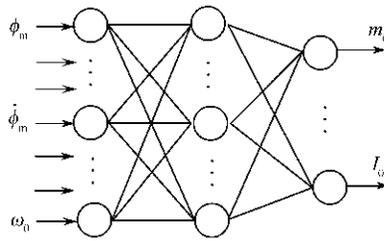


图 3-2 神经网络结构图

算法 3-2 神经网络参数辨识算法

步骤 1 首先建立神经网络训练用的样本库, 包括:

(1) 选定 FFSR 的最佳工作空间 $(\phi_{m, \text{start}}, \phi_{m, \text{End}})$, 其目的是考虑关节角限制以及尽量利用其自由飞行的特点减少训练样本范围。

(2) 计算 FFSR 本体质量 m_0 和转动惯量 I_0 的变化范围, 对给定的本体而言, 质量和转动惯量之间有着固定的关系, 因此可取 $(m_{0, \text{min}}, I_{0, \text{min}})$ 到 $(m_{0, \text{max}}, I_{0, \text{max}})$ 。以仅考虑燃料消耗为例, 本体质量和转动惯量的最小和最大值分别对应燃料用尽和最多时的情况。

(3) 改变 $\phi_{m, \text{start}} \rightarrow \phi_{m, \text{end}}, m_{0, \text{min}} \rightarrow m_{0, \text{max}}$ 和 $I_{0, \text{min}} \rightarrow I_{0, \text{max}}$ 。驱动机器人关节以给定的角速度 $\dot{\phi}_{\text{md}}$ 运动, 测量相应本体姿态变化的角速度 ω_0 , 建立关于本体惯性参数变化的样本库。

步骤 2 神经网络训练。将 $\phi_m, \dot{\phi}_{\text{md}}, \omega_0$ 作为神经网络输入, m_0 和 I_0 作为神经网络输出, 选择神经网络结构及学习算法, 进行训练, 学习式(3-10) 所描述的空间机器人的姿态干扰特性。

步骤 3 FFSR 飞行到工作区域, 进入自由浮游工作状态, 本体的惯性参数不再发生变化。此时可以纪录 ϕ_m , 驱动机器人关节以建立样本库时所设定的辨识用的角速度 $\dot{\phi}_{\text{md}}$ 运

动,姿态传感器测量此时的 ω_0 ,将 ϕ_m 、 $\dot{\phi}_{md}$ 、 ω_0 作为神经网络输入,完成前向运算,计算出本体的质量 m_0 和此时的转动惯量 I_0 。然后利用该参数构造GJM以完成运动规划和控制。

其中,步骤1和步骤2可在地面实验平台上离线完成。

该算法与利用Y. Murotsu的算法求本体的惯性参数相比,具有以下优点:①在线辨识时的计算量明显减小。在步骤3中神经网络只进行前向运算完成参数辨识,而不用建立复杂的方程组并求解,因此能够满足空间机器人对计算量和运算实时性的要求。②从算法3-2的步骤3可以看出,本算法只需测量固定关节的给定运动 $\dot{\phi}_{md}$ 时的本体姿态角速度 ω_0 ,即可求解未知的参数。不像Y. Murotsu的算法要寻找合适运动测试序列才能求解。

捕捉到惯性参数未知的目标后,却不能用上述方法,除非进行一定工程应用意义上的简化。因为此时无法建立关于未知目标惯性参数变化的样本库,它范围太广了。如果人为选定范围,则对该范围之外的目标的辨识精度就大为降低。因此针对毫无预知信息的目标,还需要进一步研究,目前可以采用Y. Murotsu的算法,尽管该算法在实际应用中还存在许多问题。

3.3.2 计算机仿真

针对双臂(每臂3个自由度)FFSR系统,进行仿真实验。设定FFSR保持 $\phi_1^2 = 0$, $\phi_3^2 = 0$, $\phi_2^2 = -\pi/4$, $\phi_1^1 = 0$, $\phi_3^1 = 0$,使 ϕ_2^1 从 -160° 变化到 140° ,步长 20° ,同时从5.0kg变化到3.0kg,测量此时的转动惯量 I_0 ,在每一状态下都驱动右臂关节2以角速度 $\dot{\phi}_2 = (\pi/4)(\text{rad} \cdot \text{s}^{-1})$ 运动,测得本体姿态角变化的角速度 ω_0 。以 ω_0 和 ϕ_2^1 作为神经网络输入, m_0 和 I_0 作为神经网络输出,测得的样本库如图3-3所示,该图对应式(3-15)的FFSR姿态干扰特性。图3-4(a)、(b)、(c)、(d)分别为常规BP算法、 $m = n = 1$ 时同伦BP算法、 $m = n = 0.5$ 时非线性同伦BP算法和非线性同伦综合(Parlos)算法迭代9500次后学习 m_0 的结果。图3-4(e)、(f)分别为 $m = n = 1$ 时同伦BP算法和非线性同伦综合(Parlos)算法迭代9500次后的学习 I_0 结果。可以看出非线性同伦Parlos算法已经收敛,具有更快的收敛速度。本章应用该数据独立运算了50次,统计结果表明 $m = n = 0.5$ 时的非线性同伦BP算法比BP算法快20倍左右,比 $m = n = 1$ 时的线性同伦BP算法快1倍左右。

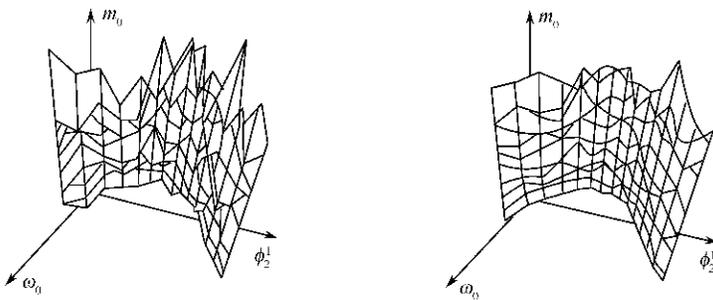


图 3-3 本体质量和转动惯量的样本库

图3-5给出了无参数辨识和参数辨识情况下的基于GJM进行规划和控制的双臂FFSR捕捉目标仿真结果。图3-5(a)为无辨识情况下(不考虑自由飞行时质量的变化,即

取本体质量恒等于 5kg) 机器人捕捉目标的运动结果。由于 FFSR 实际运动对本体的姿态

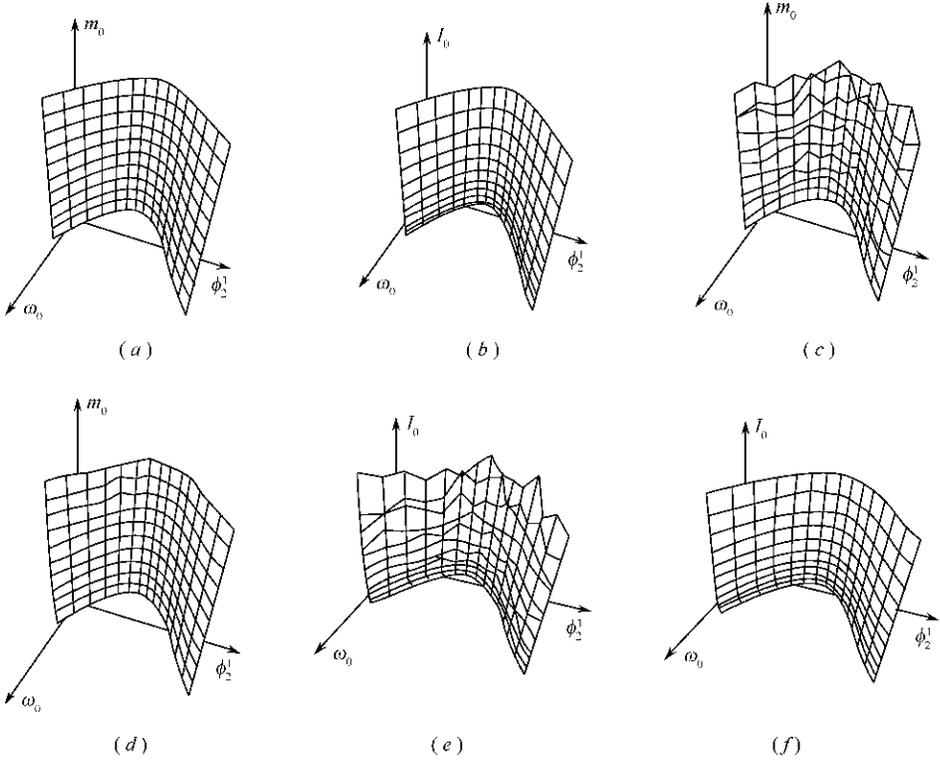


图 3-4 不同神经网络训练算法仿真结果

干扰比规划时要大, 机器人手臂末端运动轨迹产生较大偏差, 不能够准确抓住目标, 图示的例子造成机械手末端效应器与目标发生相碰。图 3-5(b) 为应用神经网络进行参数辨识的运动控制算法, 由于精确辨识了本体的质量, 利用 GJM 完成对空间机器人的运动规划和控制, 机械臂能够准确地捕捉目标。图 3-5(c) 给出了手臂末端的运动轨迹。虚线 1 为期望的理想路径, 实线 2 为无辨识情况下(即取本体质量恒等于 5kg) 机器人末端效应器运动轨迹, 实线 3 利用神经网络进行参数辨识情况下的机器人末端效应器运动轨迹。

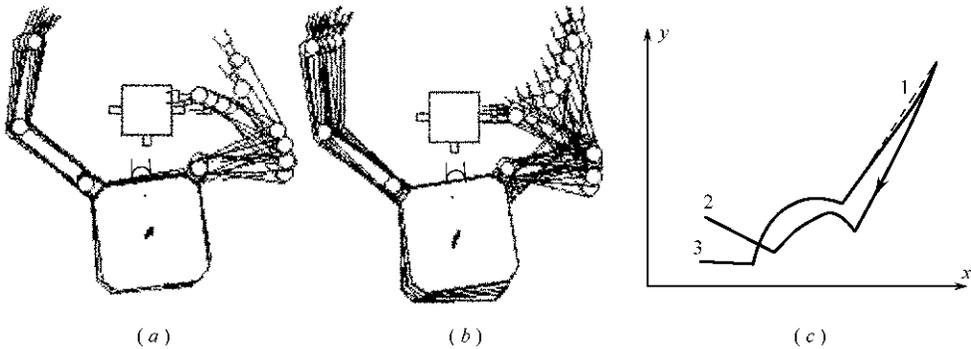


图 3-5 DFFSR 运动控制仿真结果

(a) 无辨识情况; (b) 有辨识情况; (c) 机械臂 1 末端运动轨迹。

3.4 小 结

本章在分析了 FFSR 参数辨识的必要性和可行性之后,提出了利用前馈神经网络辨识 FFSR 惯性参数的算法。考虑到 BP 算法收敛速度慢、易陷入局部极小点,提出了一种改进的同伦 BP 算法——非线性同伦 BP 算法。该算法最大的优点是其避免陷入局部极小点的能力,且在每一步迭代过程中并不增加原算法的计算复杂性。因此,注意到现有的各种改进的神经网络算法具有更快的收敛速度,又提出了非线性同伦综合算法,即用改进的算法代替非线性同伦算法中的 BP 算法,该算法在增强避免陷入局部极小能力的同时,又提高了收敛速度。在此基础上完成了 FFSR 的参数辨识,最后通过计算机仿真证明了算法的有效性。

第四章 FFSR 捕捉目标运动规划

捕捉目标是 FFSR 在空间的一项重要的作业内容,而路径规划则是研究 FFSR 如何安全、可靠并高效地捕捉目标。FFSR 工作在空间微重力的特殊环境下,这决定了 FFSR 捕捉目标运动规划的特殊性和复杂性。本章讨论 FFSR 在自由浮游状态下捕捉静态目标和动态目标的运动规划算法,提出利用 GJM 精确地控制机械手末端运动轨迹,使得 FFSR 可靠、准确地捕捉目标,并以 DFFSR 为例完成了部分仿真实验。

4.1 FFSR 运动规划

运动规划包括路径规划和轨迹规划。路径是位姿空间中的一条曲线,曲线可以用数学表达式或曲线中的点的序列表示。轨迹是对路径上的每个点指定时间,表示在某一时刻与物体相联系的位姿。路径规划与轨迹规划的区别在于,典型的路径规划仅在几何上设计机器人的位置和方向,而轨迹规划包括设计机器人运动的线速度和角速度,在路径规划中,机器人的动力学是不重要或可以忽略的。路径规划可以作为轨迹规划的第一步,在设计一个复杂的运动序列时,经常是先设计出容易的路径,然后再沿着路径分配速率。

DFFSR 运动规划的特殊是由微重力的规划环境和多臂特点决定。具体包括以下几个方面:

(1) 运动环境的特殊性。由于微重力环境下 DFFSR 的卫星本体处于自由浮游状态,没有固定基座,机械臂的运动会对卫星本体产生影响,这种影响被称为动力学干扰(Dynamics Disturbance)。因此,要研究微重力环境运动学、动力学和控制,并建立支持运动规划、支持快速实现算法的动力学干扰特征模型。

(2) 位姿空间的高维性。卫星本体的位置和姿态成为决定机器人位姿的参量,位姿空间的维数升高,以位姿空间概念为基础的规划算法的复杂度增大,问题的难度增大。

(3) 多臂协调的复杂性。FFSR 的多臂协调运动规划既要考虑操作的协调,避免碰撞的协调,还要考虑通过多臂协调运动抑制或利用微重力环境下双臂运动对卫星本体的动力学干扰。

(4) 复杂的规划内容。捕捉目标、卫星姿态控制、避免碰撞、双臂协调等都是运动规划的内容。

(5) 特殊的性能要求。由于微重力环境下物体之间的碰撞会引起物体运动状态的变化,特别是对质量较小的物体,会获得一个快速飞走的速度,这对空间操作非常不利。因此,DFFSR 运动规划对避免碰撞的安全性有很高的要求。这说明了研究微重力环境下 DFFSR 无碰撞运动规划的必要性。

4.2 FFSR 控制算法

如何控制 FFSR 的机械臂按照给定的速度、姿态跟踪给定的轨迹,是 FFSR 进行空间作业的关键技术之一。为此本节介绍基于广义雅可比矩阵(GJM)的 FFSR 机械臂运动控制算法,主要分为运动规划和运动控制两部分内容。运动规划研究 FFSR 如何安全、可靠并高效地捕捉目标,运动控制研究如何控制 FFSR 的机械臂按照规定的速度或加速度运动。目前主要有分解运动速度控制(RMRC)方法和分解运动加速度控制(RMAC)方法。

4.2.1 分解运动速度控制

分解运动速度控制是一种基于逆广义雅可比矩阵的连续路径跟踪控制方法。采用 RMRC 有两种原因:一是采用广义雅可比矩阵求解运动学问题可以线性化,仅仅通过计算逆广义雅可比矩阵就可解决。如果初始状态和关节运动的每一步都可以知道,通过数字计算就可以很容易获得惯性空间中末端执行器的位置和方向轨迹。二是考虑到采用运动速度可以很容易加入动量守恒定律,从而仅通过运动速度和动量而不考虑加速度和力就可以描述系统的反作用力效果,相当程度地简化了运算。可以有如下表达式:

$$\mathbf{v} = \mathbf{J}^* \cdot \dot{\boldsymbol{\phi}}_M \quad (4-1)$$

$$\dot{\boldsymbol{\phi}}_M = [\mathbf{J}^*]^+ \cdot \mathbf{V}_d \quad (4-2)$$

式中, \mathbf{V}_d 是所有机械臂末端效应器的线速度和角速度向量矩阵; $\dot{\boldsymbol{\phi}}_M$ 为关节角速度控制向量。由于 GJM 的推导过程已考虑了 FFSR 本体和机械手之间的相互作用,因此基于 GJM 的 RMRC 方法能够在惯性坐标系中,克服 FFSR 本体运动带来的影响,严格地控制机械臂末端效应器沿规划的路径运动。利用分解运动速度控制机械臂运动的算法如下:

算法 4-1 FFSR 分解运动速度控制算法

步骤 1 输入机器人的初始状态信息。根据机器人的几何参数和物理参数,计算机械臂的工作空间,机器人本体位置和姿态角分别为 $\mathbf{r}_0(0)$ 和 $\boldsymbol{\phi}_0(0)$, $\boldsymbol{\phi}_M(0)$ 为关节角向量。置 $i = 0$ 。

步骤 2 计算 $\dot{\boldsymbol{\phi}}_M(i) = [\mathbf{J}^*]^+ \cdot \mathbf{V}_d(i)$, 若 $|\dot{\boldsymbol{\phi}}_M(i)| > \dot{\boldsymbol{\phi}}_{\max}(i)$, 则 $\dot{\boldsymbol{\phi}}_M(i) = \text{sign}(\dot{\boldsymbol{\phi}}_M(i)) |\dot{\boldsymbol{\phi}}_{\max}(i)|$, 并给出最大速度信息, $\dot{\boldsymbol{\phi}}_{\max}$ 为机械臂最大关节角速度。

步骤 3 按照 $\dot{\boldsymbol{\phi}}_M(i)$ 驱动机械臂各关节运动。

步骤 4 计算机器人的新状态:

(1) 计算机器人本体的新位置 $\mathbf{r}_0(i+1) = \mathbf{r}_0(i) + \Delta \mathbf{r}_0$;

(2) 计算机器人本体的姿态角 $\boldsymbol{\phi}_0(i+1) = \boldsymbol{\phi}_0(i) + \Delta \boldsymbol{\phi}_0$;

(3) 计算机器人各关节角 $\boldsymbol{\phi}_M(i+1) = \boldsymbol{\phi}_M(i) + \dot{\boldsymbol{\phi}}_M(i) \Delta t$ 。

步骤 5 如果 $(i = N)$ 转步骤 6, 否则 $i = i + 1$, 转步骤 2。

步骤 6 算法结束。

在这种运动控制方法中,要考虑以下的因素:

(1) 控制周期的选择。分解速度控制方法通过在每个控制周期内输出机械臂各关节的角速度实现对机械臂的运动控制。如果控制周期选择太大,则运动控制的精度难以达到要

求;如果控制周期选择太小,在每个控制周期内都要进行费时的 J^{*-1} 的计算,给实现实时控制造成困难。因此,控制周期要选择一个适当的范围内,在地面仿真实验中一般取为 0.1s 左右。

(2) J^* 逆矩阵的求解。如果向量 $\dot{\phi}_M$ 与 V_d 的维数相等,则 J^* 是方阵,求解 J^* 的逆;否则,求 J^* 的伪逆。若 $|J^*| = 0$,则机械臂处于奇异状态,这时可以将上一个控制周期得到的 $\dot{\phi}_M$ 作为本周期的控制机械臂运动的速度值。

(3) 关节角度范围的限制。实际的机械臂系统的关节是有转动范围的,根据 RMRC 方法控制机械臂的运动应考虑这种限制条件。

(4) 关节角速度的限制。求得关节角速度可能大于相应关节角速度的限制,这时选择关节角速度范围的最大值,其他关节的角速度也按比例进行相应的调整。

4.2.2 分解运动加速度控制

如果知道末端执行器的轨迹及其每一步的运动速度,就可以利用分解运动速度控制,直接用于力矩驱动控制系统中,以完成对给定轨迹和速度的跟踪。针对式(4-1)对时间求导可得:

$$\dot{v} = \dot{J}^* \dot{\phi}_M + J^* \ddot{\phi}_M \quad (4-3)$$

整理后可得:

$$\ddot{\phi}_M = J^{*+} (\dot{v} - \dot{J}^* \dot{\phi}_M) \quad (4-4)$$

式(4-4)描述的是末端速度、关节角速度和关节角加速度间的线性关系。由于系统不受外力和动量作用,且所受的重力可以忽略不计,因此系统的动力学方程可写为:

$$\tau = H^*(\phi_M) \ddot{\phi}_M + C^*(\phi_M, \dot{\phi}_M) \quad (4-5)$$

式中, $\tau \in R^n$ 为机械臂关节控制力矩; $H^*(\phi_M) \in R^{n \times n}$ 为系统的广义惯性矩阵; $C^*(\phi_M, \dot{\phi}_M)$ 为离心力和哥氏力项。式(4-4)代入式(4-5)即可求得如下关节驱动力矩,从而驱动机械臂的运动。

$$\tau = H^*(\phi_M) J^{*+} (\dot{v} - \dot{J}^* \dot{\phi}_M) + C^*(\phi_M, \dot{\phi}_M) \quad (4-6)$$

利用式(4-6)可实现关节力矩驱动和分解速度控制。

4.2.3 机械臂典型运动分析

本章以图 2-2 提出的 FFSR 模型为例,用 FFSR 分解运动速度控制算法规划 FFSR 机械臂的运动。机械臂典型运动主要包括:

(1) 右臂末端沿着直线运动。机械臂末端沿直线运动是 FFSR 进行空间作业的最基本运动方式。首先根据直线路径规划右臂末端的运动速度 v_d^r ,利用本节的算法进行轨迹规划,求解相应的关节速度即可。仿真结果如图 4-1 所示。

(2) 右臂末端沿着直线运动,左臂末端静止。这是 FFSR 抓住目标后进行作业的基本运动方式。首先根据直线路径规划右臂末端的运动速度 v_d^r ,同时令左臂末端的运动速度 $v_d^l = 0$ 。利用式(4-2)和本节的算法进行轨迹规划,求解相应的关节速度即可。仿真结果如图 4-2 所示。

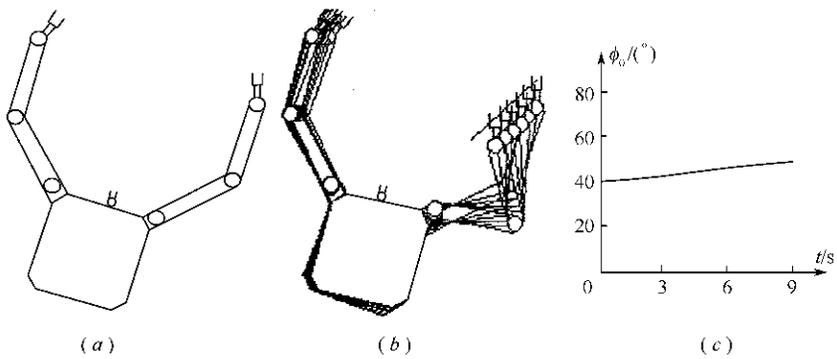


图 4-1 右臂末端沿着直线运动
(a) 初始状态; (b) 仿真结果; (c) 本体姿态变化。

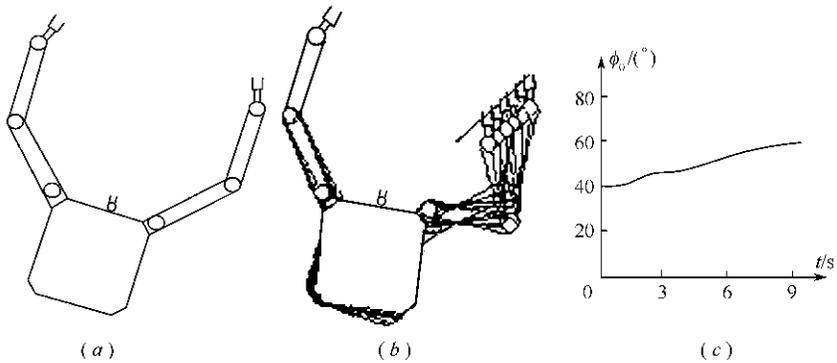


图 4-2 右臂末端沿着直线运动,左臂末端静止
(a) 初始状态; (b) 仿真结果; (c) 本体姿态变化。

(3) 右臂末端沿着圆周运动。机械臂末端沿圆周运动也是 FFSR 进行空间作业的最基本运动方式。仿真结果如图 4-3 所示。

在图 4-1 ~ 图 4-3 中,显示出机械臂运动过程中,机器人本体中心位置和本体姿态角的变化情况。这与地面机器人是不同的。

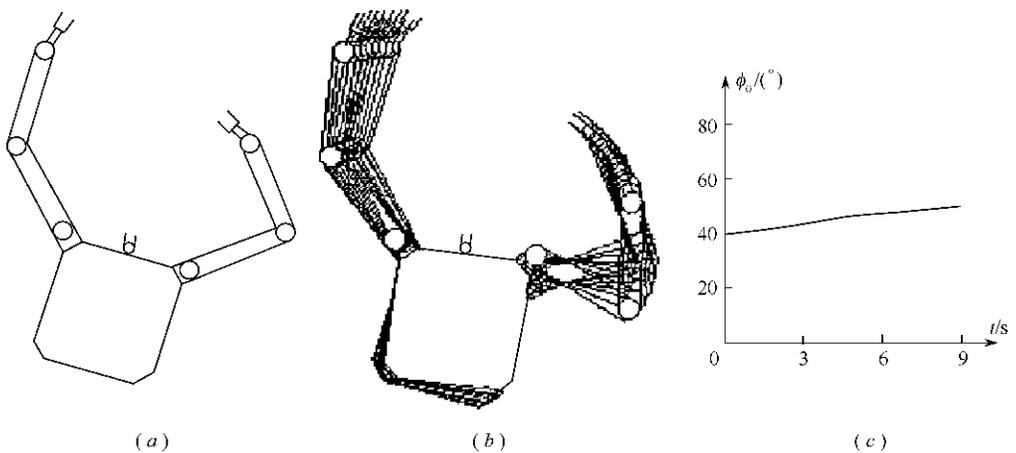


图 4-3 右臂末端沿圆周运动
(a) 初始状态; (b) 仿真结果; (c) 本体姿态变化。

4.3 FFSR 捕捉静态目标路径规划

FFSR 捕捉目标的路径规划应考虑以下几个问题:① 捕捉策略:如何减小捕捉瞬间 FFSR 和目标之间力的相互作用,使机器人能够安全、可靠地捕捉目标;② 能量消耗:由于 FFSR 携带的能量有限,应减小在 FFSR 的操作过程中消耗的能量,以延长 FFSR 的轨道寿命;③ 规划效率:由于 FFSR 的运动模型较复杂,通常需进行大量的计算,如何有效地减少计算时间,提高规划效率是 FFSR 路径规划中的一个重要问题。

4.3.1 捕捉静态目标策略

DFFSR 及目标的模型如图 4-4 所示,其中目标模型上有三只手把。当目标不受外力时,可保持静止状态或匀速运动状态。在机器人捕捉目标时,机器人的左臂捕捉目标上的手把 1,右臂捕捉目标的手把 2。机器人的本体上有一只固定臂,在机器人的双臂捕捉到目标之后,要将目标对接到本体上的固定臂。

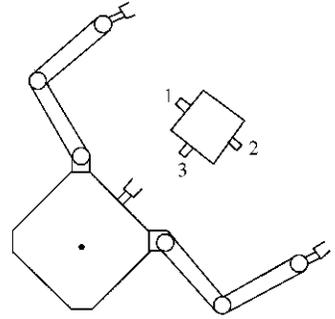


图 4-4 机器人和目标的模型

以往的 DFSR 捕捉目标的算法均采用双臂同时捕捉的策略。这种捕捉策略的控制算法比较简单,例如对于图 4-5(a) 所示的机器人和目标来说,在双臂同时捕捉的过程中,手把 1 和手把 2 的位置和姿态均固定不变,可以直接采用基于广义雅可比矩阵的分解运动速度控制方法。图(b) 中 1 为本体中心的运动轨迹,2 是左臂运动轨迹,右臂的运动轨迹用 3 表示。但双臂同时捕捉这种方法存在缺点。在双臂同时接触手把的瞬间,机械手会对目标产生挤压,因此机械手和目标之间会产生复杂的力的相互作用及动量交换,这时机器人和目标会产生无法预测的运动,这将损坏机器人和目标,有时机器人甚至会将目标推走,而无法捕捉。

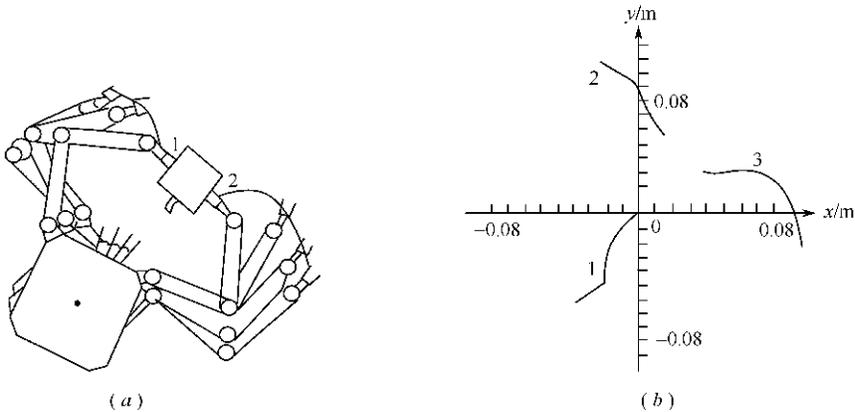


图 4-5 双臂同时捕捉目标仿真结果

(a) 捕捉过程图; (b) 目标位置变化图。

为解决这个问题,本书采用双臂先后协调捕捉的策略,将机器人捕捉目标的过程分为

3 步:

步骤 1 机器人的左臂捕捉目标上的手把 1。

步骤 2 机器人的右臂捕捉目标上的手把 2。

步骤 3 机器人用双臂协调将目标对接到机器人本体的固定臂上。

这种捕捉策略将双臂同时捕捉改为双臂先后捕捉,这种捕捉方法有效地减小捕捉瞬间机器人和目标之间力的相互作用和动量交换,保证机器人可以安全、可靠地捕捉目标。在这里,机器人的“左臂”与“右臂”是一个相对的概念,可以根据需要进行对换,因此步骤 1 与步骤 2 的顺序可以相互交换,但捕捉目标的路径规划算法都是一样的。在机器人用左臂捕捉目标上的手把 1 之后,左臂的各关节均锁定,这时左臂的最后一个连杆和目标可看作为一体,形成一个新的连杆。当机器人用右臂捕捉手把 2 时,由于右臂的运动会对机器人的本体产生反作用力,使得本体的位置和姿态发生改变,这将使目标的位置和姿态也将随之变化。因此这种捕捉算法的主要难点在于如何控制机器人的右臂捕捉比较复杂的手把 2。

4.3.2 捕捉静态目标路径规划算法

在捕捉目标的过程中采用分解运动速度(RMRC)的控制方法,RMRC 是一种基于逆雅可比矩阵的连续路径控制方法。根据式(4-3)有:

$$\dot{\phi}_M = (J^*)^+ \cdot V_d \quad (4-7)$$

式中, V_d 是所期望的机械手末端效应器的速度; $\dot{\phi}_M$ 为关节角速度控制向量; J^* 为广义雅可比矩阵(GJM)。由于 GJM 的推导过程已考虑了机器人本体和机械手之间的相互作用,因此基于 GJM 的 RMRC 方法能够在惯性坐标系中,克服机器人本体运动带来的影响,严格地控制机械手末端效应器沿所规划的路径运动,捕捉目标。

双臂 6 自由度 FFSR 左臂捕捉目标的路径规划算法如下。

算法 4-2 DFFSR 左臂捕捉目标路径规划算法

步骤 1 输入机器人的初始状态信息。根据机器人的几何参数和物理参数,计算左臂的工作空间,设目标上的手把 1 的位置为 P_t , 机器人本体位置为 $P_0(0)$, 本体姿态为 $\phi_0(0)$, $\phi_M(0)$ 为关节角向量。

步骤 2 若 P_t 不在工作空间之内,则无法捕捉,转步骤 11。

步骤 3 调节机器人的本体姿态,使 P_t 进入捕捉范围,置 $i = 0$ 。

步骤 4 计算左臂末端效应器位置 $P_c^l(i)$ 。捕捉成功,转步骤 11。

步骤 5 若 $P_c^l(i) = P_t$,则捕捉成功,转步骤 11。

步骤 6 计算 $V_d = \frac{P_t - P_c^l(i)}{\Delta t}$, Δt 为控制周期。

步骤 7 计算 $\dot{\phi}_M(i) = (J^*)^+ \cdot V_d$, 若 $|\dot{\phi}_M(i)| > \dot{\phi}_{\max}^l(i)$, 则 $\dot{\phi}_M(i) = |\dot{\phi}_{\max}^l|$, 式中 $\dot{\phi}_{\max}^l$ 为左臂最大关节运动速度。

步骤 8 按照 $\dot{\phi}_M(i)$ 驱动左臂各关节运动。

步骤 9 计算机器人的新状态:

(1) 计算机器人本体的新位置 $P_0(i+1) = P_0(i) + \Delta P_0$;

(2) 计算机器人本体的姿态角 $\phi_0(i+1) = \phi_0(i) + \Delta\phi_0$;

(3) 计算机器人各关节角 $\phi_M(i+1) = \phi_M(i) + \dot{\phi}_M(i)\Delta t$;

(4) 计算左臂末端效应器新的位置 $P_e^l(i+1)$ 。

步骤 10 $i = i + 1$, 转步骤 5。

步骤 11 算法结束。

下面对以上的算法作几点简要说明:

(1) 在步骤 7 中, 计算 $\dot{\phi}_M(i) = (J^*)^+ \cdot V_d$, 其中 $J^* \in R^{4 \times 6}$ 为广义雅可比矩阵, $(J^*)^+$ 为 J^* 的广义逆。

(2) 在步骤 9 中, $\Delta\phi_0$ 的计算可采用数值积分方法求其数值解, 本书采用了复合 Simpson 公式。

机器人的左臂捕捉目标之后, 左臂的最后一个连杆和目标可看做为一个整体, 即形成一个新的连杆。因此左臂最后一个连杆的长度、质量、转动惯量等参数发生变化, 需要新计算系统质心和广义雅可比矩阵。特别是机器人的右臂运动时, 会使机器人本体的位置的姿态改变, 使目标的位置和姿态改变, 如图 4-6 所示。因此右臂捕捉目标的路径规划算法关键在于捕捉目标上的手把 2。

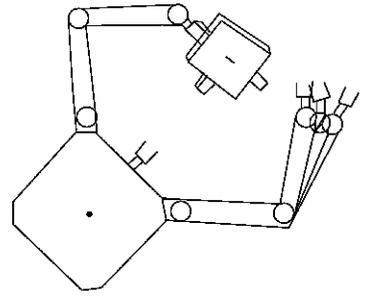


图 4-6 双臂捕捉目标后机器人及目标的状态

算法 4-3 DFFSR 右臂捕捉目标的路径规划算法

步骤 1 根据机器人的几何参数、物理参数和机器人左臂捕捉目标后的状态, 计算右臂的工作空间, 设机器人本体位置为 $P_0(0)$, 本体姿态为 $\phi_0(0)$, $\phi_M(0)$ 为关节角向量, 目标上的手把 2 的位置为 $P_t(0)$ 。

步骤 2 若 $P_t(0)$ 不在工作空间之内, 则无法捕捉, 转步骤 13。

步骤 3 $i = 0$ 。计算右臂末端效应器的位置 $P_e^r(i)$ 。

步骤 4 计算 $Distance = |P_e^r(i) - P_t(i)|$, 令 $PreDistance = Distance + 1$ 。

步骤 5 若 $Distance > PreDistance$, 则捕捉成功, 转步骤 13。

步骤 6 计算 $V_e = \frac{P_t - P_e^r(i)}{\Delta t}$, Δt 为控制周期。

步骤 7 计算 $\dot{\phi}_M(i) = (J^*)^+ \cdot V_d$, 若 $|\dot{\phi}_M(i)| > \dot{\phi}_{max}^r(i)$, 则 $\dot{\phi}_M(i) = |\dot{\phi}_{max}^r|$, 式中 $\dot{\phi}_{max}^r$ 为右臂最大关节运动速度。

步骤 8 按照 $\dot{\phi}_M(i)$ 驱动右臂各关节运动。

步骤 9 计算机器人的新状态:

(1) 计算机器人本体的新位置 $P_0(i+1) = P_0(i) + \Delta P_0$;

(2) 计算机器人本体的姿态角 $\phi_0(i+1) = \phi_0(i) + \Delta\phi_0$;

(3) 计算机器人各关节角 $\phi_M(i+1) = \phi_M(i) + \dot{\phi}_M(i)\Delta t$;

(4) 计算右臂末端效应器新的位置 $P_e^r(i+1)$ 。

步骤 10 计算目标的新状态 $P_t(i+1)$ 。

步骤 11 算法结束。 $PreDistance = Distance, Distance = |P_e^r(i+1) - P_t(i+1)|$

步骤 12 $i = i + 1$, 转步骤 5。

步骤 13 算法结束。

上面算法的关键在于计算目标的新状态 $P_1(i+1)$, 令

$$r_0^1 = b_0^1 \quad (4-8a)$$

$$r_1^1 = l_1^1 \quad (4-8b)$$

$$r_2^1 = b_2^1 \quad (4-8c)$$

$$r_3^1 = b_3^1 + \text{GoalLen} + \text{GoalGraspLen} \quad (4-9)$$

式中, GoalLen 为目标的长度; GoalGraspLen 为目标上手把的长度, 则

$$P_i(i+1) = \begin{bmatrix} P_t^x(i+1) \\ P_t^y(i+1) \end{bmatrix} = \sum_{j=0}^3 r_j^1 \cdot \begin{bmatrix} C_j \\ S_j \end{bmatrix} \quad (4-10)$$

$$C_j = \cos \left[\phi_0(i+1) + \frac{\pi}{4} + \sum_{k=1}^l \phi_k^1 \right] \quad (4-11)$$

$$S_j = \sin \left[\phi_0(i+1) + \frac{\pi}{4} + \sum_{k=1}^l \phi_k^1 \right] \quad (4-12)$$

机器人的双臂捕捉到目标后, 目标的位置可能不在机器人的对称轴上, 目标上手把 3 的方向也可能不指向机器人本体上固定臂方向, 如图 4-7 所示。这时, 应该用双臂改变目标的位置和方向, 然后再用双臂将目标对接到本体上的固定臂。

FFSR 将目标对接到固定臂上的算法与右臂捕捉目标的算法类似, 不同之处在于应同时规划左臂和右臂的运动, 以保证双臂协调运动, 将目标对接到本体上的固定臂。这里就不再给出算法了。

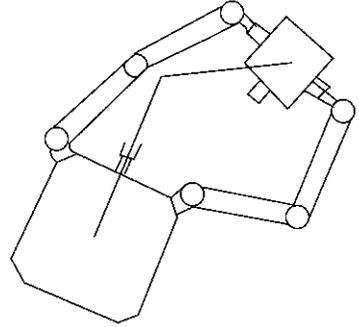


图 4-7 双臂捕捉目标后机器人及目标的状态

4.3.3 计算机仿真

DFFSR 捕捉静态目标的计算机仿真结果如图 4-8 所示。图(a)为 FFSR 和目标的初始状态, 图(b)为左臂捕捉目标上手把 1 的过程, 图(c)显示了右臂捕捉目标上手把 2 的过程, 图(d)为 FFSR 用双臂将目标对接到固定臂的过程。

图 4-9(a) 显示在捕捉过程中, FFSR 本体中心位置的变化情况, 而图(b)显示了本体姿态角的变化情况。由图中可以看出在捕捉过程中, FFSR 本体的位置和姿态均发生变化, 这与地面机器人是不同的。计算机仿真结果证明了本章提出的捕捉策略和算法是有效的。

4.4 FFSR 捕捉动态目标路径规划

4.4.1 捕捉运动目标策略

FFSR 捕捉运动目标时, 应考虑如下三个问题。

(1) 会合点的选择问题。会合点是指 FFSR 末端效应器与运动目标相遇的位置。会合点是一个空间概念, 但它的选择与时间因素是密切相关的。只有当目标运动至 FFSR 的工

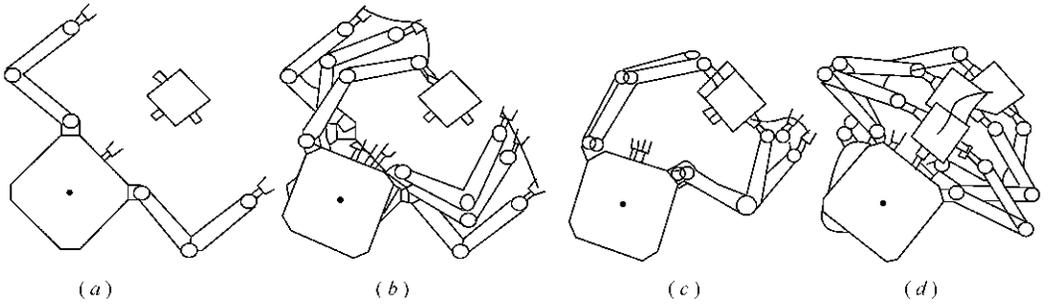


图 4-8 DFBSR 捕捉目标路径规划仿真结果
(a) 初始状态; (b) 左臂捕捉; (c) 右臂捕捉; (d) 对接。

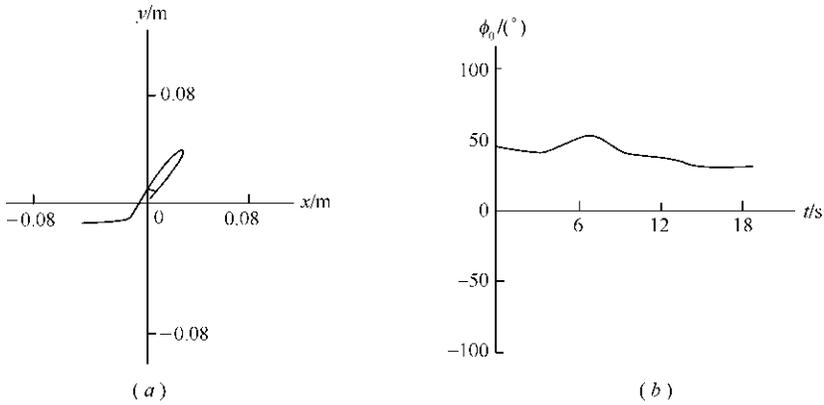


图 4-9 DFBSR 本体中心位置和姿态的变化
(a) 本体中心位置的变化; (b) 本体姿态角的变化。

作空间之内时,才能保证机器人的左臂和右臂均捕捉到目标,如图 4-10 所示。

(2) 速度问题。只有当 FFSR 末端效应器的运动速度大于目标的运动速度时,FFSR 才能成功地捕捉到目标。在空间微重力环境下,目标保持静止状态或匀速运动状态,设目标平移线速度为 V_t ,旋转角速度为 ω_t ,则末端效应器的速度应满足:

$$|V_m| > |V_t| + |R_t \cdot \omega_t| \tag{4-12}$$

式中, V_m 为末端效应器相对于固定在 FFSR 本体上的惯性坐标系的速度; R_t 为目标上的手把末端到目标中心的距离。但从安全性角度考虑,如果捕捉时末端效应器和目标的相对速度过大,会对目标和 FFSR 产生损坏,因此运动速度还应满足式(4-13),其中 V_{max} 为末端效应器和目标相对速度的最大限度。

$$|V_m| + |V_t| + |R_t \cdot \omega_t| = |V_{max}| \tag{4-13}$$

(3) 方向问题。当目标处于 FFSR 的工作空间之内,并且满足上面提出的速度要求时,仍不能保证 FFSR 捕捉到目标,如图 4-11 所示。

为满足在会合时刻,右臂末端效应器能捕捉到目标上的手把 1,目标的方向应满足下式:

$$\frac{7}{12}\pi < |\phi_r - \phi_g| < \frac{17}{12}\pi \tag{4-14}$$

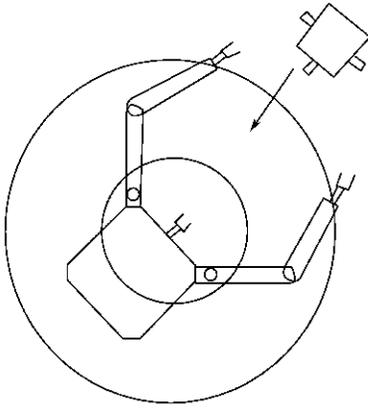


图 4-10 运动目标穿过灵活工作空间

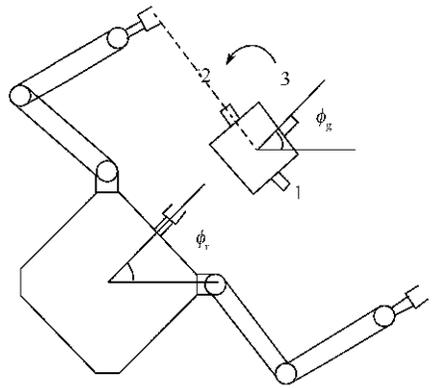


图 4-11 目标的方向不满足捕捉要求的情况

式中, ϕ_g 为目标上的手把 3 与水平线的夹角; ϕ_r 为机器人的姿态角。

4.4.2 捕捉运动目标路径规划算法

在浮游状态下捕捉运动目标时,仍采用双臂先后协调捕捉的方法,下面给出左臂捕捉运动目标的算法。右臂捕捉目标的算法和对接算法同 4.3 节中相应的算法是一致的,这是由于左臂捕捉到目标后,目标不再运动,因此 4.3 节中的算法可直接应用。

算法 4-4 DFFSR 左臂捕捉运动目标路径规划算法

步骤 1 输入机器人的初始状态信息。根据机器人的几何参数和物理参数,计算左臂的工作空间,设目标上的手把 1 的位置为 $P_t(0)$,目标的姿态角为 $\phi_g(0)$,机器人本体位置 $P_0(0)$,本体姿态为 $\phi_0(0)$, $\phi_M(0)$ 为关节角向量。

步骤 2 若目标的运动轨迹不穿过工作空间,则无捕捉,转步骤 14。

步骤 3 选择会合点 P_r 。

步骤 4 调节姿态,等待目标运动到工作空间之内。

步骤 5 $i = 0$ 。

步骤 6 计算左臂末端效应器的位置 $P_e^l(i)$ 。

步骤 7 若 $P_e^l(i) = P_r$,则捕捉成功,转步骤 14。

步骤 8 计算 $V_d = \frac{P_t - P_e^l(i)}{\Delta t}$, Δt 为控制周期。

步骤 9 计算 $\dot{\phi}_M(i) = (J^*)^+ \cdot V_d$,若 $|\dot{\phi}_M(i)| > \dot{\phi}_{\max}^l(i)$,则 $\dot{\phi}_M(i) = |\dot{\phi}_{\max}^l(i)|$,式中 $\dot{\phi}_{\max}^l$ 为左臂最大关节运动速度。

步骤 10 按照 $\dot{\phi}_M(i)$ 驱动左臂各关节运动。

步骤 11 计算机器人的新状态:

(1) 计算机器人本体的新位置 $P_0(i+1) = P_0(i) + \Delta P_0$;

(2) 计算机器人本体的姿态角 $\phi_0(i+1) = \phi_0(i) + \Delta \phi_0$;

(3) 计算机器人各关节角 $\phi_M(i+1) = \phi_M(i) + \dot{\phi}_M(i) \Delta t$;

(4) 计算左臂末端效应器新的位置 $P_e^l(i+1)$ 。

步骤 12 计算目标的新状态:

(1) 计算目标的新位置 $P_0(i+1) = P_0(i) + V_i \Delta t$;

(2) 计算目标的新姿态角 $\phi_g(i+1) = \phi_g(i) + \omega_t \cdot \Delta t$ 。

步骤 13 $i = i + 1$, 转步骤 7。

步骤 14 算法结束。

4.4.3 计算仿真

在自由浮游状态下,DFFSR 捕捉运动目标的计算机仿真结果如图 4-12 所示。图(a)为机器人和目标的初始状态,图(b)为左臂捕捉运动目标的过程仿真,图(c)为右臂捕捉运动目标的过程仿真,图(d)为机器人用双臂将目标对接到固定臂上。

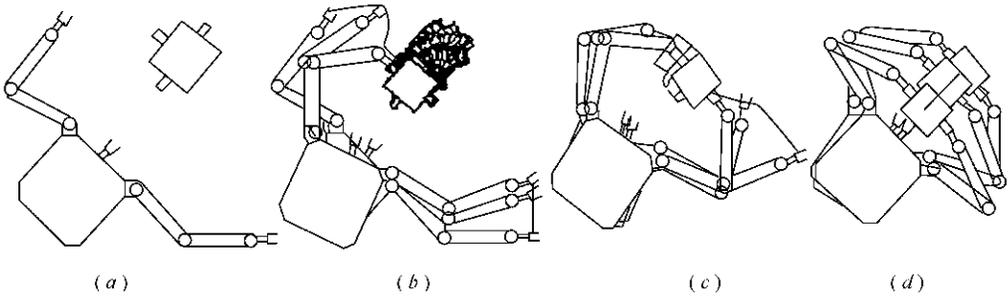


图 4-12 DFFSR 捕捉运动目标路径规划仿真结果

4.5 FFSR 避免碰撞路径规划

避免碰撞路径规划是研究如何寻找一条连接初始位置和终止位置的路径,并且这条路径不与空间中的障碍物发生重叠。由于 FFSR 工作在特殊的微重力环境,其避免碰撞规划算法比地面机器人的规划复杂。FFSR 的避免碰撞规划算法应考虑如下几个问题。

① 规划效率:为了提高路径规划的效率,应限制规划算法搜索的空间,有效地减少碰撞检测的次数,从而减少规划算法所需要的时间;② 碰撞检测:在图 4-13 所示的情况中,在右臂捕捉目标的过程中,机械手的运动会引起本体姿态的变化,使得左臂与障碍物碰撞,因此 FFSR 的碰撞检测与地面机器人的碰撞检测有所不同;③ 安全性:机械手与障碍物应保持一定的距离,以免与障碍物相碰损坏机械手。

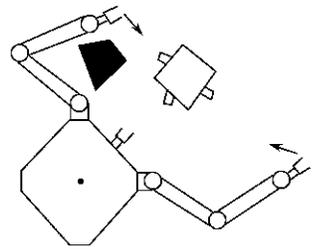


图 4-13 FFSR 的碰撞检测

4.5.1 双向双启发函数

盲目搜索工作空间的路径规划方法效率低,耗费过多的计算空间和时间。为提高路径规划效率,本章采用 A^* 算法搜索机械手的工作空间。 A^* 算法的启发式函数定义为:

$$f(x) = g(x) + h(x) \quad (4-15)$$

式中, x 为工作空间中当前待扩展的节点; $g(x)$ 为从初始节点到当前节点 x 的代价度量;

$h(x)$ 为当前节点 x 到目标节点的代价度量。

因此工作空间中任意节点的函数值 $f(x)$ 能够估算出从初始节点到当前节点 x 的最小代价路径的代价与从当前节点 x 到目标节点的最小代价路径的代价之和,也就是 $f(x)$ 是约束通过节点 x 的一条最小代价路径代价的一个估计。因此 OPEN 表中具有最小 f 值的节点就是所估计的加有最少严格约束条件的节点。不同的函数 $h(x)$ 代表不同的启发函数 $f(x)$ 。估计值 $h(x)$ 越接近实际值,函数 $f(x)$ 的搜索效率越高。函数 $g(x)$ 的定义为:

$$g(x) = \begin{cases} 0 & x = S \\ C_1(g(x') + \text{step}) + C_2\Delta\phi_0 & x \neq S \end{cases} \quad (4-16)$$

式中, x' 是 x 的父节点; step 是工作空间量化后的步长; $\Delta\phi_0$ 是机械手末端从节点 x' 运动到节点 x 时本体姿态的干扰值, C_1 和 C_2 均为常数, S 为初始节点。因此函数 $g(x)$ 考虑搜索路径的长度和机器人本体姿态干扰两个因素。

本章采用双向双启发函数搜索机器人的工作空间。双向搜索同时扩展起始节点和目标节点,当一个方向扩展进展不大时,就改变扩展方向。双启发函数利用两个不同的启发函数,在工作空间中同时搜索,规划若干步后,计算两个启发函数的效率,从中选择一个较优的函数,并以此函数进行以后的规划。这两个启发函数为:

$$\begin{cases} f_1(x) = g(x) + h_1(x) \\ f_2(x) = g(x) + h_2(x) \end{cases} \quad (4-17)$$

式中, $h_1(x)$ 和 $h_2(x)$ 可以用 $h_i(x)$ 来表示,可以写成:

$$h_i(x) = C_1 \sqrt{\sum_{j=1}^2 (M_j \times (P_j - x_j)^2)} + C_2\Delta\phi_0 \quad (4-18)$$

式中, P_j 为目标节点; x 是当前待扩展节点; $\Delta\phi_0$ 表示机械手运动对本体产生姿态干扰的估计。当 M_j 取不同的值时,代表不同的搜索方向,如 $M_j = (1, 0)$ 时,代表 x 方向优先搜索,若 $M_j = (0, 1)$ 则代表 y 方向优先搜索。搜索函数的效率可由下式计算:

$$e = C_1 \sqrt{\sum_{j=1}^2 (P_j - x_j)^2} + C_2\Delta\phi_0 \quad (4-19)$$

在式(4-19)中, e 越小说明函数的规划效率越高。

4.5.2 避免碰撞路径规划算法

算法 4-5 基于双向双启发函数的避免碰撞路径规划算法

步骤 1 将起始节点和终止节点连成一线,从起始节点开始将线上各个节点进行碰撞检测,若没有碰撞则转步骤 4。

步骤 2 将发生碰撞的第一个节点的父节点放入两个 OPEN 表中,令它们的 f 值都为零,并置两个 CLOSE 表都为空。

步骤 3 循环。若两个 OPEN 表均为空,则无解,转步骤 4。否则以双启发函数同时进行路径规划。选择 OPEN 表中具有最小值 f 的点,称为 Bestnode。若 Bestnode 为目标节点,则规划结束,转步骤 4。否则按照不同的启发函数扩展 Bestnode 节点,将 Bestnode 放入不同的 CLOSE 表中,并生成后继节点 Successor,对每个 Successor 进行如下计算:

(1) 碰撞检测,如有碰撞,则放弃此点,否则建立 Successor 到 Bestnode 的指针,并计

算

$$g(\text{Successor}) = C_1(g(\text{Bestnode}) + \text{step}) + C_2\Delta\phi_0$$

(2) 若 Successor 在 OPEN 表或 CLOSE 表中, 则称此点为 OLD, 并把它加入 Bestnode 的后继节点表中, 并比较新旧路径代价, 若 $g(\text{Successor}) < g(\text{OLD})$, 则重新确定 OLD 的父节点为 Bestnode, 并修改 $f(\text{OLD})$ 值。

(3) 若 Successor 既不在 OPEN 表中, 也不在 CLOSE 表中, 则将它放入 OPEN 表中, 并加入 Bestnode 的后继节点表中, 计算 $f(\text{Successor})$ 。

(4) 当规划次数为某一数值时, 计算双启发函数的效率, 选择效率高的函数。若此函数效率小于预先指定的阈值, 则改变搜索方向。

步骤 4 算法结束。

由于该路径规划算法将搜索过程限制在工作空间中最有希望的部分进行, 使得碰撞检测次数有效地减少, 因此该算法搜索速度快、效率高。

碰撞检测是路径规划算法的关键环节, 为减少碰撞检测所需要的时间, 将机器人的手臂抽象为线段, 机器人本体的大小也缩小相应的倍数, 同时障碍物也相应地扩展。这种方法可减少线段相交检测的次数, 因此可有效地提高碰撞检测的效率。

4.5.3 计算机仿真

为验证所提出的双向双启发函数的避免碰撞路径规划算法的可行性和有效性, 在 IBM-PC 机上做了仿真实验, 仿真结果如图 4-14 所示。

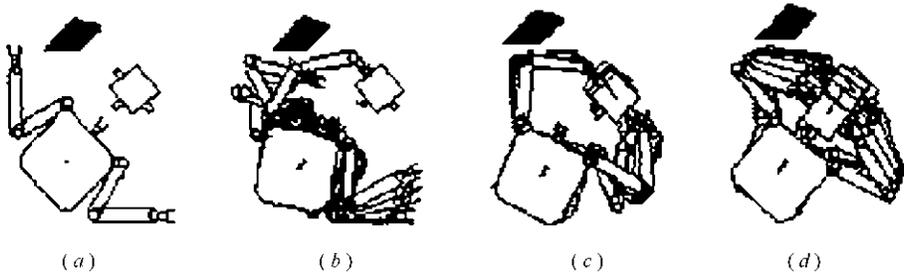


图 4-14 DFBSR 避免障碍物路径规划仿真结果
(a) 初始状态; (b) 左臂捕捉; (c) 右臂捕捉; (d) 对接。

4.6 小 结

本章讨论了 FFSR 在自由浮游状态下捕捉目标的路径规划算法。为了安全、可靠地捕捉目标, 提出了双臂先后协调捕捉的路径规划算法, 其中右臂捕捉目标的动态路径规划算法能够克服目标运动带来的不利影响, 保证右臂准确地捕捉目标。本章讨论了在自由浮游状态下, FFSR 捕捉静态目标、动态目标和避免碰撞障碍物的路径规划问题。利用双向双启发函数进行避免碰撞障碍物路径规划的研究, 该算法将搜索过程限制在工作空间中最有希望的部分进行, 有效地减少碰撞检测的次数, 提高路径规划的效率。最后通过计算机仿真实验验证了各种规划算法的可行性和正确性。

第五章 基于受限最小干扰图的姿态控制

在空间微重力环境下,搭载在 FFSR 本体上的机械手的运动会对本体产生反作用力和力矩,使 FFSR 本体的位置和姿态发生变化。FFSR 为完成复杂的空间作业,必须保证本体内携带的通信系统正常工作,因此需要保证本体上搭载的天线固定地指向某一方向。例如,地面的通信站或轨道上的通信卫星等。因此姿态控制问题是 FFSR 的一个重要研究领域,姿态控制主要研究如何保证 FFSR 的本体姿态在机械臂的运动过程中保持不变,或者,按照给定方式改变本体及机械臂的姿态,使它们达到预先指定的角度。本章主要阐述基于受限最小干扰图的姿态控制方法。

5.1 FFSR 的姿态控制方法

目前,对姿态控制的研究可以分为三类:① 利用机器人本体内携带的反作用轮和反作用喷气装置来保证本体的姿态或位置不变;② 设计特殊几何构造的机械臂,利用对称性、平衡机械臂运动对本体产生的反作用力和力矩,使机器人本体的姿态保持不变;③ 通过精确地控制机械臂的运动轨迹,使得机器人的本体姿态保持不变,或按指定要求转动。

日本的 K. Yoshida 提出了基于计算动量的反作用补偿方法(Computed Momentum Based Reaction Compensation, CMRC)。通常传统的基于反馈的姿态控制方法是在传感器检测到姿态误差后,再调整姿态。这种方法在实际应用中不是很有效。因此 K. Yoshida 提出了带有预测功能的前馈式控制策略 CMRC 方法。CMRC 方法是基于动量守恒的速度层次上的前馈式控制方法,它利用动量守恒定律计算由于机械臂运动使本体产生的动量,然后控制反作用轮转动,抵消机械臂运动带来的影响,使姿态保持不变,这种方法要比基于计算力矩的姿态控制方法简单,需要的计算量也较小,日本的 ETS-Ⅶ 型机器人卫星采用这种姿态控制方式。此外 R. Longman 等人, M. W. Walker 和 L. B. Wee 等人都提出了不同的方法利用反作用轮或反作用喷气装置进行姿态控制。反作用喷气装置的使用将消耗机器人携带的宝贵的不可再生的燃料,减小机器人轨道使用寿命;而反作用轮调节姿态的能力有限,当反作用轮的能量达到饱和时,必须将能量转移,否则反作用轮就无法连续工作。因此在姿态控制中应尽量少用反作用喷气装置和反作用轮。

E. Papadopoulos 和 A. Abu-Abad 已证明对于自由浮游状态下的 FFSR,存在着对 FFSR 的姿态不产生干扰的路径。这时 FFSR 的本体只会平移,而不会转动。对于平面机械手,如果遵循如下设计原则,则会使 FFSR 的本体在机械手运动过程中既不产生平移,也不产生旋转。① 采用旋转关节和对称结构的连杆,使机械手的受力平衡,从而抵消机械手运动对本体产生的反作用力;② 利用质量矩阵的不变性和特殊的规划技术,使机械手对本体产生的反动量为零。满足上述要求的 FFSR 一定为冗余机械手。为验证上述理论,作

者设计了一个 3 自由度 9 连杆的并行机械手,它的 3 个驱动电机都安装在本体上,而且共享同一个转动轴。当电机转动方向相反时,可以抵消电机转动对本体产生的动量,保证机械手的运动对本体不产生干扰。采用这种方法进行姿态控制的主要缺点在于机械手的构造很复杂,不易控制。

Z. Vafa 和 S. Dubowsky 提出了一个自调节姿态控制方法,它利用 FFSR 所特有的非完整性,通过机械手在关节空间作闭合路径的运动调节本体的姿态角。当机械手在关节空间中沿某一路径从初始位置出发,再沿另一路径回到初始位置时,机器人本体方向就会改变。这种方法的缺点是:① 它只是一个近似算法,因为它假设机械手沿相同的闭合路径运动,本体姿态改变的角度都是一样的,但实际上 FFSR 本体姿态改变的大小是不同的;② 姿态调节效率低,即使本体姿态只需改变很小的角度,也需要做很多次闭合路径运动,因此姿态调节效率低。

S. Dubowsky 和 M. A. Torres 在基于 FFSR 动力学特点的基础上,提出了利用增强干扰图(Enhanced Disturbance Map, EDM)方法用以规划机器人的运动。机械手按照不同的方向运动对本体姿态的影响也不同,产生最大姿态干扰的方向和最小姿态干扰的方向在关节空间中是相互垂直的,因此控制机械手的运动方向可有效地减小本体姿态的变化。利用 EDM 进行姿态控制的不足之处在于:计算 EDM 需要的计算量很大,而且占用较多的存储空间。当机械手沿规划得到的路径运动时,这种方法不能保证本体的姿态不发生改变,因此这种方法需消耗燃料以维持 FFSR 本体姿态不变。

P. Nenchev 提出的基于固定姿态受限雅可比矩阵(Fixed-Attitude-Restricted Jacobian Matrix, FARJM)的姿态控制方法,控制机械手的末端沿某一路径运动,同时使 FFSR 的姿态按要求的方式改变,或使姿态改变最小。这种方法在求广义雅可比矩阵的伪逆时,同时考虑机械手惯性矩阵空间影响,并在规划过程中引入李雅普诺夫函数,保证在控制机械手运动的同时,控制 FFSR 的本体姿态。这种方法可采用分解速度和分解加速度两种控制方式。它的优点是避开了机械手的奇异点,缺点是计算量较大。

C. Fernandes 和 L. Gurvits 等人提出的姿态控制算法主要解决两种问题:① 通过机械手的运动调节 FFSR 本体的姿态,但机械手的初始姿态和终止姿态保持不变;② 如何使 FFSR 的本体姿态保持不变,而改变机械手的姿态。根据动量守恒定律和动能定理,首先建立机械手的动力学模型,在此基础上作者提出一个递归算法,以解决 FFSR 的可控性问题。即给定任何两个位姿 x_0 和 x_f ,如何求解一个位姿序列连接 x_0 和 x_f 。

K. Yoshida 等人研究多臂 FFSR 的运动规律,提出利用某一臂的运动来补偿另一臂的运动对机器人本体姿态产生的影响。采用这种姿态控制方法时,一个臂根据指定的路径运动,通过另一个臂的运动保持本体姿态的稳定,并优化整个系统的操作力矩。此外俄亥俄大学的 S. K. Agrawal 和 S. Shirumalla, H. Fujii 等人、Y. Chen 等人也提出了类似的方案。

Y. Nakamura 和 R. Mukherjee 讨论了 FFSR 在自由浮游状态下特有的非完整性。他们利用角动量守恒将姿态控制问题转化为一个非线性控制问题。控制的输入是机械手的关节角速度,状态变量为本体的姿态参数和机械手关节角。采用李雅普诺夫函数,利用双向规划的方法,可以同时控制机械手的位置和姿态。但是对于一个非完整系统,一般存在平滑的反馈,这种控制方法不能保证机器人以指定的姿态到达某些点。

此外,也可将其他学科中的控制理论应用到姿态控制方法中。如 Senda 等人利用神经网络

络进行姿态控制。M. Sampei 和 H. Kiyota 等人将控制论中的理论应用到姿态控制中。FFSR 是一个典型的非完整系统,在某些情况下,即使系统可控时,通过连续的静态状态反馈,也不能使系统在平衡位置稳定。Sampei 等人提出一个新的基于时间状态控制表 (Time-State Control Form) 的姿态控制方法。任何一个非完整系统可以模型化为 m 个输入, n 个状态的非线性系统,该系统包括两个状态方程:一个是状态控制方程,另一个是时间控制表,它控制第一个方程的时间流。对一个 FFSR,可以将本体的角速度和线速度作为系统的输入变量,而将各关节角作为系统状态,用坐标变换和输入变换可以将空间机器人的状态方程变成一个时间状态控制表,增加或者减少时间控制表中的时间标尺,系统的状态输出也随之改变,然后再将线性静态状态反馈增益作为新的输入,如此逐渐逼近所期望的状态。

5.2 干扰图与增强干扰图

5.2.1 干扰图

Z. Vafa 和 S. Dubowsky 提出干扰图 (Disturbance Map, DM) 的方法用来描述机械手的运动对机器人本体姿态产生的干扰。对于一个二连杆的单臂机械手,它的干扰图 (DM) 如图 5-1 所示。

图 5-1 中的每条线段代表 FFSR 最小姿态干扰运动轨迹,线段的长度代表姿态干扰值的大小,箭头的方向为最小姿态干扰的方向。干扰图的缺点在于:

- (1) 无法将最小姿态干扰运动轨迹以清晰的方式描述出来,因此 DM 的可视性不好;
- (2) 干扰图无法描述多于二连杆的机械手的姿态变化关系。

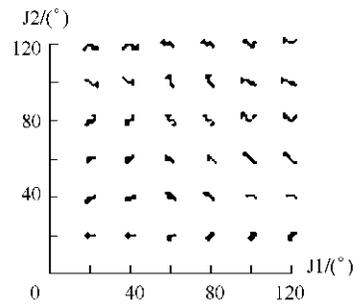


图 5-1 干扰图

5.2.2 增强干扰图

为将干扰图进行改进,S. Dubowsky 和 M. A. Torres 于 1991 年提出了基于增强干扰图 (EDM) 的姿态控制方法。假设机器人的本体上搭载一个 n 自由度的机械手,本体中心位置为 $\mathbf{X}_b = [x, y, z]^T$,本体姿态角用 $\Theta_b = [\phi, \theta, \psi]^T$ 表示,机械手关节角向量为 $\mathbf{q} = [q_1, q_2, \dots, q_n]^T$,令向量 $\xi = [\mathbf{X}_b, \Theta_b, \mathbf{q}]^T$,定义广义力 $\mathbf{F} = [f_x, f_\theta, f_q]^T$,式中 f_x 和 f_θ 为反作用力在机器人本体中心产生的力和力矩, f_q 为作用在机械手各关节的力和力矩。则系统的运动方程可写为:

$$\mathbf{H}(\xi)\ddot{\xi} + \mathbf{C}(\xi, \dot{\xi})\dot{\xi} = \mathbf{F} \quad (5-1)$$

式中, $\mathbf{H}(\xi) \in R^{(n+6) \times (n+6)}$ 为对称正定惯性矩阵; $\mathbf{C}(\xi, \dot{\xi}) \in R^{(n+6) \times (n+6)}$ 是与离心力有关的矩阵; $\mathbf{F} \in R^{(n+6) \times 1}$ 是广义力向量。

系统的广义动量为:

$$\boldsymbol{\pi} = \mathbf{H}(\xi)\dot{\xi} \quad (5-2)$$

式中, $\boldsymbol{\pi} = [\pi_x, \pi_y, \pi_z, \pi_\phi, \pi_\theta, \pi_\psi, \pi_1, \pi_2, \dots, \pi_n]^T$,其中 π_x, π_y, π_z 为本体中心的线动量; $\pi_\phi,$

π_θ, π_ψ 为本体中心的角动量; $\pi_1, \pi_2, \dots, \pi_n$ 是机械手各关节在旋转轴上的动量分量。

假设没有外力和力矩作用于系统, 系统的初始动量为零, 并且不用反作用喷气装置控制系统的位置和姿态, 联立式(5-1) 和式(5-2), 则式(5-1) 中的 $H(\xi)$ 可写成:

$$H(\xi) = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \quad (5-3)$$

式中, $A \in R^{6 \times 6}$ 是描述机器人线速度和角速度与系统线动量和角动量的矩阵; $B \in R^{6 \times n}$ 是描述关节运动与系统线动量和角动量的质量矩阵; $C = B^T$; $D \in R^{n \times n}$ 是描述关节角速度与关节动量的矩阵。

将式(5-3) 代入式(5-2) 中, 得:

$$[\dot{X}_b^T, \dot{\Theta}_b^T]^T = -A^{-1} \dot{B}_q \quad (5-4)$$

令 $Z = [z_1, z_2]^T = -A^{-1} B$, 式中 $z_1 \in R^{3 \times n}$ 和 $z_2 \in R^{3 \times n}$, 则式(4-4) 可写成:

$$\begin{bmatrix} \delta X_b \\ \delta \Theta_b \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \delta q \quad (5-5)$$

则

$$\delta \Theta_b = z_2 \delta q \quad (5-6)$$

将式(5-6) 进行奇异值分解:

$$z_2 = UDV^T \quad (5-7)$$

式中, U 和 V 都为正交矩阵, U 代表对本体产生最小姿态干扰方向的矩阵, V 为对本体产生最大姿态干扰方向的矩阵; $D \in R^{6 \times n}$ 为对角阵。将 z_2 进行奇异值分解后, 可分别得到机械手运动对本体产生最大姿态干扰的方向和最小姿态干扰的方向, 若将所有最大干扰方向和最小干扰方向分别连接起来, 则可得到反映机械手运动对本体姿态产生干扰的最大姿态干扰曲线和最小姿态干扰曲线。FFSR 的姿态干扰曲线不仅与机器人的几何参数有关, 而且还与机器人的物理参数如质量等有关。

对于双臂 4 自由度 FFSR, 其本体和机械手的姿态如图 5-2(a) 所示, 则它左臂的增强干扰图(EDM) 如图(b) 所示。

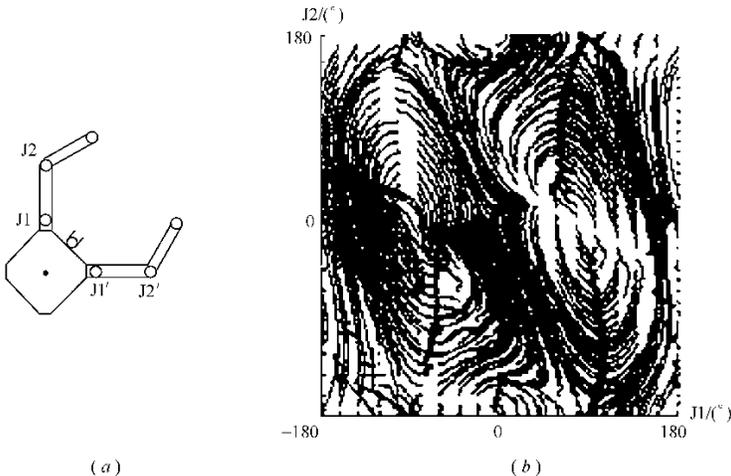


图 5-2 4 自由度 DFFER 的模型和左臂增强干扰图(2 自由度)

(a) 双臂 4 自由度 FFSR; (b) 增强干扰图(2 自由度)。

对于双臂 6 自由度 FFSR,其本体和机械手的姿态如图 5-3(a) 所示,它左臂的 EDM 如图(b) 所示。

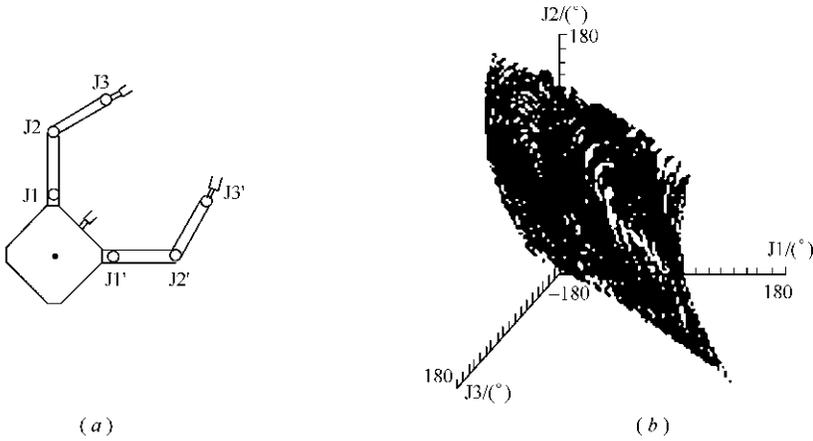


图 5-3 6 自由度 DFFSR 的模型和左臂增强干扰图(3 自由度)

(a) 双臂 6 自由度 FFSR; (b) 增强干扰图(3 自由度)。

EDM 图的缺点在于:

- (1) 计算增强干扰图(EDM) 要消耗大量的时间;
- (2) 存储增强干扰图(EDM) 要消耗大量的空间;
- (3) 不能保证机械手的运动对本体产生的干扰为零。

5.3 姿态干扰计算及受限最小干扰图

5.3.1 姿态干扰计算

本书的第二章推导了描述机械手末端运动速度与各关节运动速度关系的广义雅可比矩阵,本节在此基础上,推导出机械手各关节运动速度与本体角速度的关系。对于图 2-1 所示的 FFSR,其右臂末端效应器的线速度为:

$$\dot{p}^r = v_0 + \omega_0 \times (p^r - r_0) + \sum_{i=1}^m [u_i^r \times (p^r - d_i^r)] \cdot \dot{\phi}_i^r \quad (5-8)$$

式中, v_0 为机器人本体的线速度; ω_0 为机器人本体的角速度; p^r 为机器人右臂末端效应器的位置向量; u_i^r 为右臂第 i 个关节的旋转轴的单位向量; d_i^r 为右臂第 i 个关节的位置向量; $\dot{\phi}_i^r$ 为右臂第 i 个关节的角速度。同理,左臂末端效应器的线速度 \dot{p}^l 可写成:

$$\dot{p}^l = v_0 + \omega_0 \times (p^l - r_0) + \sum_{i=1}^n [u_i^l \times (p^l - d_i^l)] \cdot \dot{\phi}_i^l \quad (5-9)$$

右臂末端效应器的角速度为:

$$\omega^r = \omega_0 + \sum_{i=1}^m u_i^r \cdot \dot{\phi}_i^r \quad (5-10)$$

左臂末端效应器的角速度 ω^l 可写成:

$$\omega^l = \omega_0 + \sum_{i=1}^n u_i^l \cdot \dot{\phi}_i^l \quad (5-11)$$

令, $\mathbf{v}^r = [(\dot{\mathbf{p}}^r)^\top, (\boldsymbol{\omega}^r)^\top]^\top$, $\mathbf{v}^l = [(\dot{\mathbf{p}}^l)^\top, (\boldsymbol{\omega}^l)^\top]^\top$, 并将 \mathbf{v}^r 和 \mathbf{v}^l 写成通用表示形式:

$$\mathbf{v}^c = [(\dot{\mathbf{p}}^c)^\top, (\boldsymbol{\omega}^c)^\top]^\top \quad (5-12)$$

当 $c = r$ 时, 代表右臂末端效应器的速度向量; 当 $c = l$ 时, 代表左臂末端效应器的速度向量, 则 \mathbf{v}^c 可写成下面的形式:

$$\mathbf{v}^c = \mathbf{J}_S^c \begin{bmatrix} \mathbf{v}_0 \\ \boldsymbol{\omega}_0 \end{bmatrix} + \mathbf{J}_M^c \dot{\boldsymbol{\phi}}_M \quad (5-13)$$

式中, $\mathbf{J}_S^c \in R^{6 \times 6}$ 为与 FFSR 本体相关的雅可比矩阵; $\mathbf{J}_M^c \in R^{6 \times (m+n)}$ 为与机械手相关的雅可比矩阵; $\boldsymbol{\phi}_M$ 为 FFSR 各关节的角速度, $\boldsymbol{\phi}_M = (\phi_1^r, \dots, \phi_m^r, \phi_1^l, \dots, \phi_n^l)^\top$ 。在式(5-13)中,

$$\mathbf{J}_S^c = \begin{bmatrix} \mathbf{E}_{3 \times 3} & -\bar{\mathbf{P}}_{r0}^c \\ \mathbf{O}_{3 \times 3} & \mathbf{E}_{3 \times 3} \end{bmatrix} \quad (5-14)$$

当 $c = r$ 时,

$$\mathbf{J}_M^r = \begin{bmatrix} \mathbf{u}_1^r \times (\mathbf{p}^r - \mathbf{d}_1^r) & \mathbf{u}_2^r \times (\mathbf{p}^r - \mathbf{d}_2^r) & \cdots & \mathbf{u}_m^r \times (\mathbf{p}^r - \mathbf{d}_m^r) & \mathbf{O}_{3 \times n} \\ \mathbf{u}_1^r & \mathbf{u}_2^r & \cdots & \mathbf{u}_m^r & \mathbf{O}_{3 \times n} \end{bmatrix} \quad (5-15)$$

当 $c = l$ 时,

$$\mathbf{J}_M^l = \begin{bmatrix} \mathbf{O}_{3 \times m} & \mathbf{u}_1^l \times (\mathbf{p}^l - \mathbf{d}_1^l) & \mathbf{u}_2^l \times (\mathbf{p}^l - \mathbf{d}_2^l) & \cdots & \mathbf{u}_m^l \times (\mathbf{p}^l - \mathbf{d}_m^l) \\ \mathbf{O}_{3 \times m} & \mathbf{u}_1^l & \mathbf{u}_2^l & \cdots & \mathbf{u}_m^l \end{bmatrix} \quad (5-16)$$

在式(5-14) ~ 式(5-16)中, $\mathbf{E}_{3 \times 3}$ 为 3×3 单位矩阵, $\mathbf{O}_{s \times t}$ 代表 $s \times t$ 的零矩阵。当 $c = r$ 时,

$$\mathbf{p}_{r0}^r = \mathbf{p}^r - \mathbf{r}_0 \quad (5-17)$$

当 $c = l$ 时,

$$\mathbf{p}_{r0}^l = \mathbf{p}^l - \mathbf{r}_0 \quad (5-18)$$

对于任意一个向量 $\mathbf{r} = (r_x, r_y, r_z)^\top$, 定义 $\bar{\mathbf{r}}$ 为:

$$\bar{\mathbf{r}} = \begin{bmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{bmatrix} \quad (5-19)$$

系统的总线动量 Ω 和总角动量 Θ 分别为:

$$\Omega = m_0 \dot{\mathbf{r}}_0 + \sum_{i=1}^m m_i^r \cdot \dot{\mathbf{r}}_i^r + \sum_{i=1}^n m_i^l \cdot \dot{\mathbf{r}}_i^l \quad (5-20)$$

$$\Theta = I_0 \boldsymbol{\omega}_0 + m_0 \mathbf{r}_0 \times \dot{\mathbf{r}}_0 + \sum_{i=1}^m (I_i^r \boldsymbol{\omega}_i^r + m_i^r \mathbf{r}_i^r \times \dot{\mathbf{r}}_i^r) + \sum_{i=1}^n (I_i^l \boldsymbol{\omega}_i^l + m_i^l \mathbf{r}_i^l \times \dot{\mathbf{r}}_i^l) \quad (5-21)$$

将式(5-20)和式(5-21)写成 \mathbf{v}_0 、 $\boldsymbol{\omega}_0$ 和 $\dot{\boldsymbol{\phi}}_M$ 的函数, 则有:

$$\begin{bmatrix} \Omega \\ \Theta \end{bmatrix} = \begin{bmatrix} \mathbf{W} \mathbf{E}_{3 \times 3} & -\mathbf{W} \bar{\mathbf{r}}_{g0} \\ \mathbf{W} \bar{\mathbf{r}}_g & \mathbf{I}_t \end{bmatrix} \begin{bmatrix} \mathbf{v}_0 \\ \boldsymbol{\omega}_0 \end{bmatrix} + \begin{bmatrix} \mathbf{J}_M \\ \mathbf{I}_M \end{bmatrix} \dot{\boldsymbol{\phi}}_M \quad (5-22)$$

式中, W 为系统的总质量。

$$W = m_0 + \sum_{i=1}^m m_i^r + \sum_{i=1}^n m_i^l \quad (5-23)$$

$$\mathbf{r}_{g0} = \mathbf{r}_g - \mathbf{r}_0 \quad (5-24)$$

上述式中, r_g 为系统质心的位置向量; r_i^r 为右臂连杆 i 质心的位置向量; r_i^l 为左臂连杆 i 质心的位置向量。

$$I_t = I_0 + \sum_{i=1}^m (I_i^r - m_i^r \bar{r}_i^r \bar{r}_{i0}^r) + \sum_{i=1}^n (I_i^l - m_i^l \bar{r}_i^l \bar{r}_{i0}^l) \quad (5-25)$$

式(5-25)中 r_{i0} 为:

$$r_{i0} = r_i - r_0 \quad (5-26)$$

式(5-22)中的 J_M 为:

$$J_M = \sum_{i=1}^m m_i^r J_{Mi}^r + \sum_{i=1}^n m_i^l J_{Mi}^l \quad (5-27)$$

式中

$$J_{Mi}^r = \begin{bmatrix} u_1^r \times (r_i^r - d_1^r) & u_2^r \times (r_i^r - d_2^r) & \cdots \\ u_i^r \times (r_i^r - d_i^r) & O_{3 \times 3} & \cdots & O_{3 \times 3} \end{bmatrix} \quad (5-28)$$

$$J_{Mi}^l = \begin{bmatrix} O_{3 \times 3} & \cdots & O_{3 \times 3} & u_1^l \times (r_i^l - d_1^l) & u_2^l \times (r_i^l - d_2^l) & \cdots \\ u_i^l \times (r_i^l - d_i^l) & O_{3 \times 3} & \cdots & O_{3 \times 3} \end{bmatrix} \quad (5-29)$$

$$I_M = \sum_{i=1}^m (I_i^r J_{Si}^r + m_i^r \bar{r}_i^r J_{Mi}^r) + \sum_{i=1}^n (I_i^l J_{Si}^l + m_i^l \bar{r}_i^l J_{Mi}^l) \quad (5-30)$$

式中

$$J_{Si}^r = [u_1^r \quad u_2^r \quad \cdots \quad u_i^r \quad O_{3 \times 1} \quad \cdots \quad O_{3 \times 1}] \quad (5-31)$$

$$J_{Si}^l = [O_{3 \times 1} \quad \cdots \quad O_{3 \times 1} \quad u_1^l \quad u_2^l \quad \cdots \quad u_i^l \quad O_{3 \times 1} \quad \cdots \quad O_{3 \times 1}] \quad (5-32)$$

假设没有外力及力矩作用于系统,并且系统的初始总动量为零,则有:

$$v_0 = \left(-\bar{r}_{g0} I_S^{-1} I_m - \frac{1}{W} J_M \right) \cdot \dot{\phi}_M = J_v \dot{\phi}_M \quad (5-33)$$

$$\omega_0 = -I_S^{-1} I_m \dot{\phi}_M = J_\omega \dot{\phi}_M \quad (5-34)$$

式中, $I_S \in R^{3 \times 3}$ 为机器人的惯性矩阵; $I_m \in R^{3 \times n}$ 为机械手的惯性矩阵。

$$I_S = I_t + W \bar{r}_g \bar{r}_{g0} \quad (5-35)$$

$$I_m = I_M - \bar{r}_g J_M \quad (5-36)$$

将式(5-35)和式(5-36)代入式(5-13)中,消去 v_0 和 ω_0 ,得:

$$v^c = J_S^c \begin{pmatrix} v_0 \\ \omega_0 \end{pmatrix} + J_M^c \dot{\phi}_M = \left[\begin{pmatrix} J_v + \bar{P}_{r0} J_\omega \\ J_\omega \end{pmatrix} + J_M^c \right] \dot{\phi}_M = J^c \dot{\phi}_M \quad (5-37)$$

式(5-37)中, J^c 描述了机械手的运动速度和本体姿态角速度间的关系,也就是描述了机械手的运动对机器人本体姿态产生的干扰。式(5-37)中的 J^c 与式(5-6)中的 z_2 实质上是一致的,但由于式(5-37)在推导过程中仅需求解一阶微分方程,因此 J^c 的计算比 z_2 的计算简单。

5.3.2 受限最小干扰图

若计算图 5-2 所示的增强干扰图(EDM),并将它存储起来,既要消耗很多的计算时间,又要浪费很多的存储空间。为了提高计算 EDM 的效率,并减少存储 EDM 所需的空间,本节将第二章中工作空间的计算应用到计算 EDM 的算法中,并将惯性坐标系中的目标映射到关节空间中,从而有效地限制计算 EDM 时所需搜索的空间,并节省计算时间。

按照这种方法得到的新曲线称为受限最小干扰图 (Restricted Minimum Disturbance Map, RMDM)。计算 RMDM 的算法如下。

算法 5-1 计算左臂 RMDM 算法

步骤 1 输入 FFSR 各连杆的质量、长度、转动惯量等参数。

步骤 2 计算 FFSR 的工作空间,并将目标映射到关节空间中。设左臂关节 1 的最小关节角为 $J1Start$,左臂关节 1 的最大关节角为 $J1End$,左臂关节 2 的最小关节角为 $J2Start$,左臂关节 2 的最大关节角为 $J2End$ 。

步骤 3 令 $i = J1Start, j = J2Start$ 。

步骤 4 若 $i > J1End$,并且 $j > J2End$,则算法结束,转步骤 9。

步骤 5 计算当前姿态的 $J^1(i, j)$ 。

步骤 6 将 $J^1(i, j)$ 进行奇异值分解,得到最小姿态干扰方向。

步骤 7 $j = j + step2$ 。

步骤 8 (1) 若 $i < J1End$,并且 $j > J2End$,则

$$i = i + step1; j = J2Start。$$

(2) 转步骤 4。

步骤 9 算法结束。

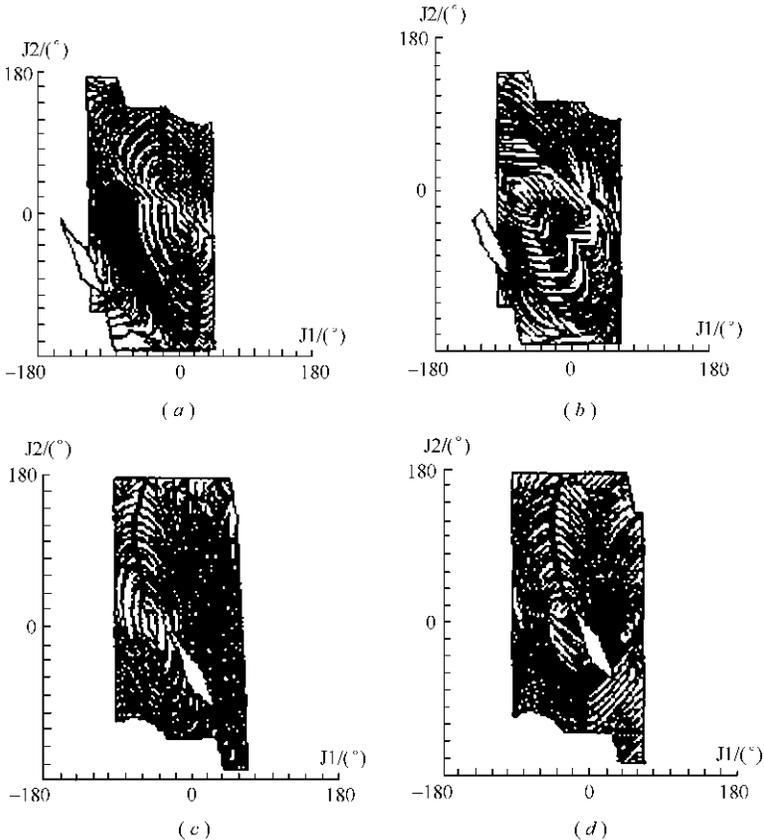


图 5-4 4 种受限干扰图

(a) 左臂受限最小干扰图; (b) 左臂受限最大干扰图; (c) 右臂受限最小干扰图; (d) 右臂受限最大干扰图。

上述算法中步骤 7 和步骤 8 中的 step1 和 step2 是关节角 1 和关节角 2 量化后变化的步长。按上述算法计算得到的受限最小干扰图(RMDM) 如图 5-4(a) 所示。同理可以得到双臂 4 自由度 FFSR 的左臂受限最大干扰图(Restricted Maximum Disturbance Map, RMADM) 和右臂的受限最小干扰图和受限最大干扰图。它们分别如图(b)、图(c) 和图(d) 所示。在图 5-4 显示的 4 种干扰图中, 由实线围成的轮廓代表机器人的工作空间, 图形中间的空白区域是目标在关节空间中的映射, 图中的每条曲线代表机器人的运动路径, 其中的间断点是机器人的奇异点。

同理将图 5-3 中的双臂 6 自由度 FFSR 的增强干扰图(EDM) 按照算法 5-1 提出的方法重新计算, 得到 FFSR 的左臂 3 自由度受限最小干扰图(RMDM), 其结果如图 5-5 所示。

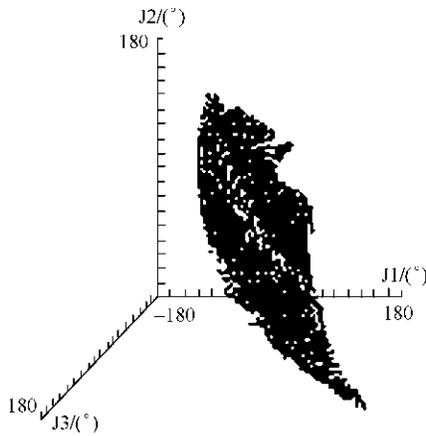


图 5-5 左臂 RMDM(3 自由度)

5.3.3 RMDM 图与 EDM 图的比较

式(5-6) 描述了关节角运动速度与本体角速度的关系, 它是计算 EDM 图所需的运动模型, 它与式(5-37) 实质上是等价的。但由于在式(5-6) 的推导过程中需要求解二阶微分方程, 而式(5-37) 只需求解一阶微分方程, 因此式(5-6) 的计算要比公式(5-37) 的计算复杂得多, 即采用式(5-6) 中得到的模型计算增强干扰图(EDM) 要消耗大量的计算时间。

如果采用算法 5-1 提出的计算 RMDM 方法计算最小姿态干扰曲线, 所需的计算时间和存储空间都比 EDM 图要小。算法 5-1 计算得到的结果, 可以用双向链表结构表示。在该结构中记录各关节角的姿态、本体姿态的干扰值、指向下一节点的指针和指向前一节点的指针等, 采用这种结构可以存储各种姿态特性曲线。它的数据结构定义如下:

```
struct Node
{
    float Joint1, Joint2, Joint3;           // 左(右) 臂关节角的姿态
    float PreAttitudeDisturbance;         // 前向姿态干扰值
    float NextAttitudeDisturbance;       // 后向姿态干扰值
    struct Node * PreNode;                // 指向前一节点的指针
}
```

```

struct Node * NextNode;           // 指向后一节点的指针
}
    
```

图 5-6 和图 5-7 分别表示了计算不同的姿态干扰图所需的时间和存储空间。两个图中符号定义是一致的。1 表示计算 2DOF 左臂 EDM 所需的时间, 2 表示计算 2DOF 左臂 RMDM 所需要的时间, 3 表示计算 3DOF 左臂 EDM 所需的时间, 4 表示计算 3DOF 左臂 RMDM 所需的时间。从图中可以看出, 计算 RMDM 所需的时间和存储 RMDM 所需的空间都比 EDM 图所需的时间和空间要小。这是由于: ① 采用的模型不同, EDM 图的计算需求解二阶微分方程, 而计算 RMDM 只需计算一阶微分方程; ② 将工作空间的概念引入计算 RMDM 中, 有效地限制算法 5-1 中需要搜索的关节空间。

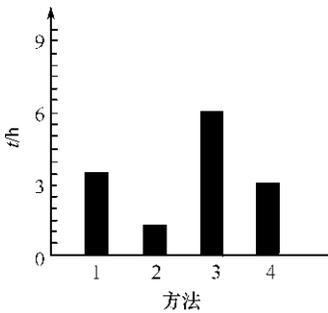


图 5-6 计算时间的比较

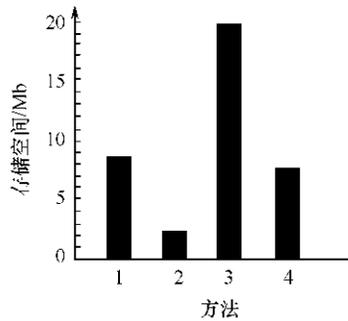


图 5-7 存储空间的比较

综上所述, 受限最小干扰图与增强干扰图相比, 它的优点在于: ① 采用较简单的运动模型: 计算 EDM 所需的运动模型, 需多次求解二阶微分方程, 而 RMDM 只需计算一阶微分方程, 有效地减少数值积分的次数; ② 计算时间短: 由于运动模型简单, 并将工作空间的概念引入计算 RMDM 算法中, 因此计算 RMDM 所需的时间比计算 EDM 所需的时间短; ③ 存储空间小: 由于将工作空间的概念应用在计算 RMDM 的算法中, 并将目标映射到关节空间中, 有效地限制存储 RMDM 所需的空间。

5.4 基于 RMDM 的姿态控制算法

S. Dubowsky 提出的姿态控制算法, 计算的时间长, 占用空间大, 而且采用这种方法控制机器人操作时, 不能保证机械手运动对本体不产生干扰。因此本节提出一个新的改进算法, 这种方法需要的计算时间较短, 占用空间了比 S. Dubowsky 提出的方法小, 而且这种方法可以保证机械手的运动对本体姿态不产生干扰, 因此消耗的能量也小。

5.4.1 零姿态干扰的计算

式(5-34)中 $\dot{\phi}_M$ 为 FFSR 各关节角速度的向量, 它可写成:

$$\dot{\phi}_M = (\dot{\phi}_1^r, \dot{\phi}_2^r, \dots, \dot{\phi}_m^r, \dot{\phi}_1^l, \dot{\phi}_2^l, \dots, \dot{\phi}_n^l)^T \tag{5-38}$$

将其代入式(5-34), 有:

$$\omega_0 = J_\omega (\dot{\phi}_1^r, \dot{\phi}_2^r, \dots, \dot{\phi}_m^r, \dot{\phi}_1^l, \dot{\phi}_2^l, \dots, \dot{\phi}_n^l)^T \tag{5-39}$$

为保证机械手的运动对 FFSR 的本体姿态不产生干扰,只需令上式中的 ω_0 为零,即:

$$\mathbf{J}_\omega(\dot{\phi}_1^r, \dot{\phi}_2^r, \dots, \dot{\phi}_m^r, \dot{\phi}_1^l, \dot{\phi}_2^l, \dots, \dot{\phi}_n^l)^T = 0 \quad (5-40)$$

因此当机械手的运动满足式(5-40)时,即可保证机械手的运动对 FFSR 的本体姿态不产生干扰。由于本书采用双臂 6 自由度 FFSR,式(5-40)可简写成:

$$\mathbf{J}_\omega(\dot{\phi}_1^r, \dot{\phi}_2^r, \dot{\phi}_3^r, \dot{\phi}_1^l, \dot{\phi}_2^l, \dot{\phi}_3^l)^T = 0 \quad (5-41)$$

5.4.2 基于 RMDM 的姿态控制算法

本节提出的姿态控制算法利用了 5.3 节的受限最小干扰图(RMDM),并在此基础上采用有效的搜索策略,使得机器人能够快速、准确地捕捉到目标,同时保证机械手的运动对本体产生的姿态干扰最小。如果机器人本体的姿态干扰不为零,按照式(5-41)规划右臂的运动,保证机器人既能捕捉到目标,又可以满足机械手的运动对本体姿态不产生干扰。

算法 5-2 基于 RMDM 的姿态控制算法

步骤 1 输入机器人和目标的初始状态。设机器人的本体位置为 $P_0(0)$,本体的姿态角为 $\phi_0(0)$,机械手各关节角向量为 $\phi_M(0)$,目标的位置为 P_t ,目标的姿态为 ϕ_g 。

步骤 2 读入左臂受限最小干扰图(RMDM)。

步骤 3 将机器人的初始姿态映射到关节空间中,设此点为 CNode,并将其放入 OPEN 表中。将目标节点映射到关节空间中,设此点为 GNode。置 CLOSE 表为空。置 FORWARD = 0。

步骤 4 循环。若 OPEN 表为空,则失败退出。转步骤 6。否则选择 OPEN 表中具有最小 f 值的点称为 Best Node。若 Best Node 为目标节点,则捕捉成功,转步骤 6。否则将 Best Node 放入 CLOSE 表中,并进行如下计算。

(1) 计算 CNode. PreNode 和 CNode. NextNode 的姿态角。

若 $\text{Distance}(\text{CNode. PreNode}, \text{GNode}) > \text{Distance}(\text{CNode}, \text{GNode})$,

则置 FORWARD = 1;

否则若 $\text{Distance}(\text{CNode. NextNode}, \text{GNode}) > \text{Distance}(\text{CNode}, \text{GNode})$

则置 FORWARD = -1;

(2) 若 PNode 与 GNode 在同一姿态干扰曲线上,则转步骤 5。

(3) 生成 BestNode 的后继节点 Successor,若 Successor 在 OPEN 表或 CLOSE 表中出现过,则放弃此点,否则将此点放入 OPEN 表中,计算 $f(\text{Successor}) = g(\text{Successor}) + h(\text{Successor})$ 。

(4) 计算右臂协调运动的速度值 $(\dot{\phi}_1^r, \dot{\phi}_2^r, \dot{\phi}_3^r)$ 。

步骤 5 若 CNode = GNode,则捕捉成功,转步骤 6。

否则

若 FORWARD = 1,则 CNode = CNode. PreNode;

若 FORWARD = -1,则 CNode = CNode. NextNode。

计算右臂协调运动的速度值 $(\dot{\phi}_1^r, \dot{\phi}_2^r, \dot{\phi}_3^r)$ 。

步骤 6 算法结束。

下面对上面算法做一下分析和说明。

(1) 算法 5-2 所描述的姿态控制算法是在关节空间中进行的,因此在步骤 4 中的函数 Distance(CNode, CNode) 所计算的距离为关节空间中当前节点与目标节点之间的距离;

(2) 在步骤 3 和步骤 4 中提出了启发式搜索算法,该算法的目的是使当前搜索节点快速地与目标节点在同一条最小姿态干扰曲线上。本书采用算法搜索机器人的关节空间,在式 $f(\text{Successor}) = g(\text{Successor}) + h(\text{Successor})$ 中, Successor 为当前待扩展节点的后继节点, $g(\text{Successor})$ 是从初始节点到 Successor 的代价度量, $h(\text{Successor})$ 是从 Successor 到目标节点的代价估计。其中估计函数 $h(x)$ 为当前节点与目标节点的距离和手臂运动对本体姿态干扰的函数,它的定义如下:

$$h(x) = C_1 \sqrt{\sum_{i=1}^3 (\text{GNode. Joint}_i - \text{CNode. Joint}_i)^2} + C_2 \Delta\phi_0 \quad (5-42)$$

在式(5-42)中, C_1 和 C_2 为常数; $\Delta\phi_0$ 为机械手运动对本体产生的干扰,它等于本体当前姿态和运动前姿态的差值。选择不同的估计函数 $h(x)$, 启发式函数 $f(x)$ 也就不同,因此搜索的效率也不同。

(3) 在步骤 4 中计算 PNode 是否与 GNode 在同一条最小姿态干扰曲线上的算法为:

算法 5-3 任意两个节点是否在同一条最小姿态干扰曲线上的判定算法

步骤 1 输入两个节点 Node1 和 Node2。

步骤 2 计算 PreDistance = Distance(Node1, Node2)。

步骤 3 循环:

① if(FORWARD == 1)

{

if(Node1. PreNode == Node2),

则 Node1 与 Node2 在同一条干扰曲线上,转步骤 5;

else Node1 = Node1. PreNode;

}

else if(FORWARD == -1)

{

if(Node1. NextNode == Node2),

则 Node1 与 Node2 在同一条干扰曲线上,转步骤 5;

else Node1 = Node1. NextNode;

}

② 计算 CurrentDistance = Distance(Node1, Node2);

③ 若 CurrentDistance > PreDistance,则转步骤 4;

否则 PreDistance = CurrentDistance。

步骤 4 Node1 和 Node2 不在同一条姿态干扰曲线上。

步骤 5 算法结束。

(4) 在步骤 4 和步骤 5 中,当 FFSR 左臂的运动使本体姿态发生变化时,应通过右臂协调运动,使 FFSR 的本体姿态保持不变。右臂的运动规划可以依据式(5-41),其协调控

制算法如下。

算法 5-4 右臂协调运动姿态控制算法

步骤 1 输入左臂三个关节的运动速度 $(\dot{\phi}_1^l, \dot{\phi}_2^l, \dot{\phi}_3^l)$ 。

步骤 2 计算 $\omega_0 = J_w(0, 0, 0, \dot{\phi}_1^l, \dot{\phi}_2^l, \dot{\phi}_3^l)$ 。

步骤 3 若 ω_0 为零,则转步骤 6。

步骤 4 计算 $J_w(\dot{\phi}_1^r, \dot{\phi}_2^r, \dot{\phi}_3^r, \dot{\phi}_1^l, \dot{\phi}_2^l, \dot{\phi}_3^l) = 0$ 。令 $\dot{\phi}_3^l$ 为零,对任意指定的 $\dot{\phi}_1^r$ 计算 $\dot{\phi}_2^r$ 的值。

步骤 5 输出 $(\dot{\phi}_1^r, \dot{\phi}_2^r, \dot{\phi}_3^r)$,控制关节运动。

步骤 6 算法结束。

在上面的算法中,右臂的控制方法不是惟一的,只需保证 FFSR 各关节运动的连续性,任意的解都是可行的。为减小右臂运动所需的能量,令右臂的最后一个关节保持不动,而通过前两个关节的运动调节机器人的本体姿态保持不变。这种右臂协调的方法可有效地节省能量。

5.4.3 计算机仿真

为验证本节提出的基于 RMDM 姿态控制算法的可行性及效率,作者在 PC 机上做了如下三个实验。首先不采用姿态控制算法控制机器人捕捉目标,其仿真结果如图 5-8 所示。其中,图(a)为机器人和目标的初始状态,捕捉目标的运动过程如图(b)所示,图(c)为末端效应器的运动轨迹。从仿真结果中可以看出在机械手的运动过程中,尽管机器人捕捉到目标,但本体的姿态发生了很大的变化。

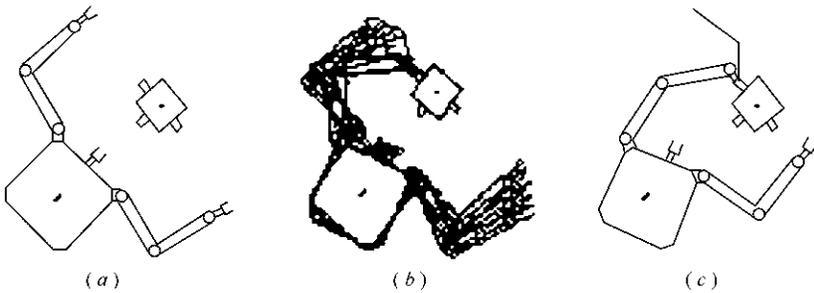


图 5-8 在无姿态控制状态下 FFSR 捕捉目标的仿真结果

(a) 初始状态; (b) 捕捉状态; (c) 末端轨迹。

当 FFSR 和目标的初始姿态和图 5-8 所示的状态相同时,采用 S. Dubowsky 提出的姿态控制算法控制 FFSR 捕捉目标,其仿真结果如图 5-9 所示。图(a)为初始状态,图(b)为捕捉目标的运动过程,图(c)为末端效应器的运动轨迹。从图中可以看出,FFSR 本体的姿态也发生了变化,但姿态变化要比无姿态控制的情况要小。

如果 FFSR 和目标的初始姿态仍与图 5-8 一致,采用基于受限最小干扰图(RMDM)的姿态控制算法,控制 FFSR 捕捉目标,并通过右臂的协调运动,使机器人的本体姿态保持不变,其仿真结果如图 5-10 所示。图(a)为机器人和目标的初始状态,图(b)为机器人捕捉目标及右臂协调的运动过程,图(c)为左臂和右臂末端效应器的运动轨迹。从图中可以

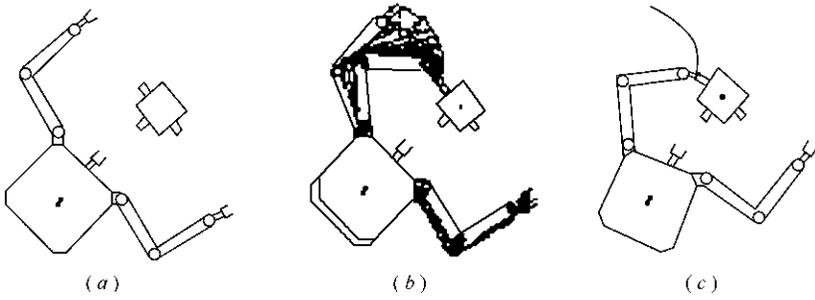


图 5-9 利用 S. Dubowsky 提出的姿态控制算法捕捉目标的仿真结果
(a) 初始状态; (b) 捕捉状态; (c) 末端轨迹。

看出, 尽管机器人的本体位置发生了漂移, 但其姿态却保持不变。

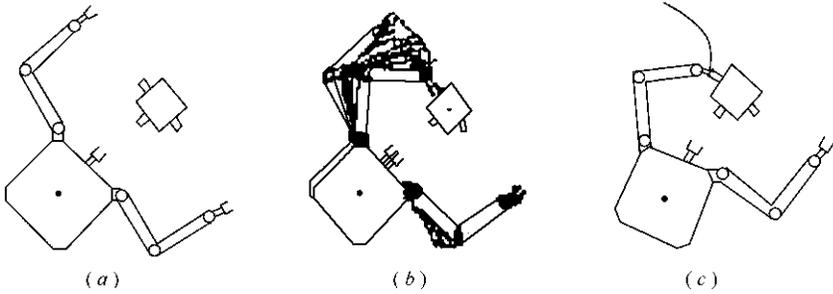


图 5-10 使用基于受限最小干扰图姿态控制方法捕捉目标的仿真结果
(a) 初始状态; (b) 捕捉状态; (c) 末端轨迹。

在上面三个计算机仿真实验中, FFSR 本体姿态的变化如图 5-11 所示。图中曲线 1 表示无姿态控制时捕捉目标的情况, 曲线 2 代表利用 S. Dubowsky 提出的姿态控制算法捕捉目标时本体姿态的变化过程, 曲线 3 是采用本节提出的基于最小姿态干扰曲线 (RMDM) 的姿态控制算法捕捉目标时机器人本体变化情况。从图中可以看出, 方法 1 中的姿态干扰最大, 方法 2 中的姿态干扰较小, 而方法 3 中无姿态变化。

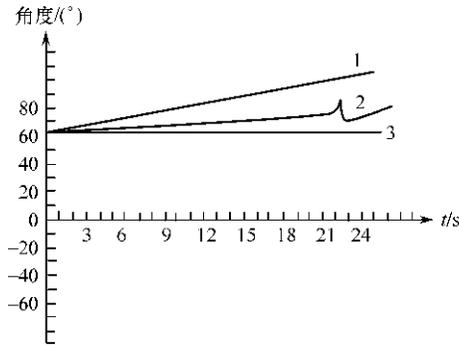


图 5-11 各种算法仿真结果的比较

由于采用基于受限最小干扰图 (RMDM) 的姿态控制算法在进行姿态控制时, FFSR 的本体姿态不变, 因此不需消耗能量来使反作用轮或反作用喷气装置工作, 以维持本体姿

态不变,因此可节省能量,延长 FFSR 的轨道寿命。从仿真结果比较中,可以得到如下结论:基于 RMDM 的姿态控制算法计算速度快,占用内存空间小,消耗燃料也比其他两种方法小。

5.5 小 结

本章讨论了双臂 FFSR 的姿态控制问题。首先综述了各种姿态控制方法,并指出了它们存在的优点和缺点。其次推导了双臂 FFSR 机械手运动和本体角速度的关系,在此基础上,改进了 S. Dubowsky 提出的计算 EDM 图方法,提出了计算受限最小干扰图(RMDM)的方法,计算 RMDM 所需要的时间以及存储 RMDM 所需要的空间都比 EDM 图所需要的少。最后提出基于 RMDM 的姿态控制算法,这种算法采用有效的启发式搜索策略,使得 FFSR 能快速、准确地捕捉到目标,同时保障 FFSR 本体的姿态不发生变化。并通过计算机仿真实验验证了所提出的姿态控制算法的正确性和可行性。

第六章 基于姿态稳定的冗余 FFSR 运动规划

为了保证 FFSR 的通信系统和电源系统的正常工作,必须确保 FFSR 搭载的碟形天线和太阳能帆板的定向,因此姿态稳定对 FFSR 而言是至关重要的,即应该保证 FFSR 本体的姿态在机械臂运动过程中保持不变。为此,针对冗余 FFSR,本章提出了关于多臂 FFSR 末端效应器任务空间的姿态受限广义雅可比矩阵概念,并利用该矩阵完成了捕捉目标的协调运动规划,使 FFSR 在完成空间作业的同时,通过多关节的协调运动,相互抵消对本体的姿态干扰。该算法不但计算量较小,而且可以用于存在有空间障碍物等其他约束环境中的运动规划,是一种适应性更强的建模方法。

6.1 基于姿态稳定的冗余多臂 FFSR 运动分析

目前,许多文献研究基于姿态稳定的 FFSR 完成作业任务时的运动规划问题,但是多数是针对单臂或双臂 FFSR,对冗余多臂 FFSR 的研究比较少。本节重点考虑冗余多臂 FFSR,该机器人具有足够的自由度,使得其在驱动机械臂的末端效应器跟踪给定路径的同时,对本体的姿态不产生干扰,分析此时机械臂末端效应器的运动速度和关节角速度的关系。

6.1.1 FFSR 冗余性分析

可以把 FFSR 的运动分为两类:FFSR 本体的运动和机械臂末端效应器的运动。这些运动决定了相对于惯性坐标系的本体和末端效应器的位置和姿态。定义 m_1 和 m_2 分别代表本体和末端效应器的任务空间。

系统总的自由度则为: $n = \sum_{k=1}^l n_k$, n_k 是第 k 个机械臂的自由度。因此,系统总的自由度 n 和 FFSR 任务空间的关系有如下三种情况。

(1) $n > m_1 + m_2$, 此时机械臂运动相对于本体和末端效应器的运动而言都是冗余的。此时,不但可以用来控制末端效应器和本体的协调运动,还有额外的冗余,可以用来满足某些附加的要求,如关节角及角速度的限制、避开奇异点和避碰等。

(2) $n = m_1 + m_2$, 此时仅能满足本体和机械臂的协调运动,不具有额外的冗余,用来避碰、避开奇异点等。

(3) $n < m_1 + m_2$, 则不能够保证在机械臂末端跟踪给定任务的同时维持本体姿态的稳定。因此,本章仅考虑前两种情况。

6.1.2 姿态受限广义雅可比矩阵

考虑图 2-2 提出的多臂 FFSR 的模型。对于这个系统,期望能够协调机械臂末端效应

器的运动(v^k)和本体姿态角速度(ω_0),即在机械臂的运动时可得多臂 FFSR 的姿态干扰特性和第 k 个机械臂的末端速度和 FFSR 各关节角速度的关系方程:

$$\mathbf{I}_m \dot{\boldsymbol{\phi}}_m = 0 \quad (6-1)$$

$$\mathbf{v}^k = \mathbf{J}_m^k \dot{\boldsymbol{\phi}}_m \quad (6-2)$$

因此,这里根据不同的应用情况,给出以下两种不同的分析方法。

1. 多关节协调运动

将 FFSR 的关节空间看为一个整体,而不分为各个机械臂的关节角进行考虑。当第 k 个机械臂完成给定任务时,在整个关节空间进行多关节协调运动,使得 FFSR 工作时不对本体产生姿态干扰。由式(6-1)利用冗余伪逆公式得:

$$\dot{\boldsymbol{\phi}}_m = (\mathbf{I} - \mathbf{I}_m^+ \mathbf{I}_m) \dot{\boldsymbol{\theta}} \quad \forall \boldsymbol{\theta} \in R^n \quad (6-3)$$

式中,任意矢量 $\dot{\boldsymbol{\theta}}$ 可由其他的约束来求得。最终的目的是规划第 k 个机械臂沿着给定的轨迹运动时,对本体不产生干扰。将式(6-3)代入式(6-2)可得:

$$\mathbf{v}^k = \mathbf{J}_m^k (\mathbf{I} - \mathbf{I}_m^+ \mathbf{I}_m) \dot{\boldsymbol{\theta}} \quad (6-4)$$

从式(6-4)中求出 $\dot{\boldsymbol{\theta}}$ 则有:

$$\dot{\boldsymbol{\theta}} = [\mathbf{J}_m^k (\mathbf{I} - \mathbf{I}_m^+ \mathbf{I}_m)]^+ \mathbf{v}^k \quad (6-5)$$

将式(6-5)代入式(6-3)可得:

$$\dot{\boldsymbol{\phi}}_m = (\mathbf{I} - \mathbf{I}_m^+ \mathbf{I}_m) [\mathbf{J}_m^k (\mathbf{I} - \mathbf{I}_m^+ \mathbf{I}_m)]^+ \cdot \mathbf{v}^k \quad (6-6)$$

利用矩阵公式 $P(HP)^+ = (HP)^+$,其中 P 是对称、幂等矩阵和投影算子。由式(6-6)可得:

$$\dot{\boldsymbol{\phi}}_m = [\mathbf{J}_m^k (\mathbf{I} - \mathbf{I}_m^+ \mathbf{I}_m)]^+ \cdot \mathbf{v}^k = [\mathbf{J}_{AR}^{*k}]^+ \mathbf{v}^k \quad (6-7)$$

式中

$$\mathbf{J}_{AR}^{*k} \equiv \mathbf{J}_m^k (\mathbf{I} - \mathbf{I}_m^+ \mathbf{I}_m) \quad (6-8)$$

此处称 \mathbf{J}_{AR}^{*k} 为 FFSR 第 k 个机械臂的基于整个关节空间的姿态受限广义雅可比矩阵(ARGJM), \mathbf{J}_{AR}^{*k} 则代表了姿态稳定时第 k 个机械臂末端效应器的运动速度和关节角速度的关系。因此,如果利用该矩阵完成多臂 FFSR 的分解运动速度控制,即保证了机械臂运动不会对 FFSR 本体的姿态产生干扰,从而完成机械臂和本体的协调运动。

2. 多臂协调运动

将 FFSR 的关节空间看为由各个机械臂的关节角组成。当第 k 个机械臂完成给定任务时,利用其他的机械臂进行协调运动,使得 FFSR 工作时不对本体产生姿态干扰。由式(6-1)和式(6-2)分别可得:

$$\mathbf{I}_m \dot{\boldsymbol{\phi}}_m = \mathbf{I}_m^1 \dot{\boldsymbol{\phi}}_m^1 + \mathbf{I}_m^2 \dot{\boldsymbol{\phi}}_m^2 + \cdots + \mathbf{I}_m^l \dot{\boldsymbol{\phi}}_m^l = 0 \quad (6-9)$$

$$\mathbf{v}^k = \mathbf{J}_m^k \dot{\boldsymbol{\phi}}_m = \mathbf{J}_m^{k,1} \dot{\boldsymbol{\phi}}_m^1 + \mathbf{J}_m^{k,2} \dot{\boldsymbol{\phi}}_m^2 + \cdots + \mathbf{J}_m^{k,l} \dot{\boldsymbol{\phi}}_m^l \quad (6-10)$$

式(6-9)和式(6-10)分别可写为:

$$\mathbf{I}_m^k \dot{\boldsymbol{\phi}}_m^k + \mathbf{I}_m^{1-k} \dot{\boldsymbol{\phi}}_m^{1-k} = 0 \quad (6-11)$$

$$\mathbf{v}^k = \mathbf{J}_m^{k,k} \dot{\boldsymbol{\phi}}_m^k + \mathbf{J}_m^{k,1-k} \dot{\boldsymbol{\phi}}_m^{1-k} \quad (6-12)$$

式中, \mathbf{I}_m^k 和 $\mathbf{J}_m^{k,k}$ 分别为矩阵 \mathbf{I}_m 和 \mathbf{J}_m^k 中第 k 个机械臂的关节角所对应的部分所形成的子矩阵, \mathbf{I}_m^{1-k} 和 $\mathbf{J}_m^{k,1-k}$ 为矩阵 \mathbf{I}_m 和 \mathbf{J}_m^k 中去掉第 k 个机械臂的关节角所对应的部分所形成的子矩阵。由式(6-11)可得:

$$\dot{\boldsymbol{\phi}}_m^{1-k} = -[\mathbf{I}_m^{1-k}]^+ \mathbf{I}_m^k \dot{\boldsymbol{\phi}}_m^k \quad (6-13)$$

代入式(6-12)经整理后可得:

$$\mathbf{v}^k = (\mathbf{J}_m^{k,k} - \mathbf{J}_m^{k,1-k}[\mathbf{I}_m^{1-k}]^+ \mathbf{I}_m^k) \dot{\boldsymbol{\phi}}_m^k = \mathbf{J}_{AR}^k \dot{\boldsymbol{\phi}}_m^k \quad (6-14)$$

式中

$$\mathbf{J}_{AR}^k \equiv \mathbf{J}_m^{k,k} - \mathbf{J}_m^{k,1-k}[\mathbf{I}_m^{1-k}]^+ \mathbf{I}_m^k \quad (6-15)$$

\mathbf{J}_{AR}^k 称为 FFSR 第 k 个机械臂的基于多臂协调的姿态受限广义雅可比矩阵 (ARGJM)。可以先利用 \mathbf{J}_{AR}^k 在第 k 个机械臂的关节空间内规划该机械臂的运动, 然后利用式(6-13)求解其他机械臂的协调运动, 即可实现基于姿态稳定的 FFSR 的运动规划。

3. 不同方法比较

这两种方法都能够完成基于姿态稳定的多臂 FFSR 的运动规划, 而在具体的应用中, 对于这两种方法的优劣, 可以考虑以下两方面的因素。

(1) 从计算量方面考虑。 \mathbf{J}_{AR}^k 与 \mathbf{J}_{AR}^{*k} 相比计算量要小得多, 这是由于在进行机械臂的运动规划时, 求 $\mathbf{J}_{AR}^{*k} \in R^{m_2 \times n}$ 的伪逆比求 $\mathbf{J}_{AR}^k \in R^{m_2 \times n_k}$ 的伪逆的计算量要大得多。

(2) 从协调运动效果来考虑。在进行 FFSR 设计时, 如果有意识地区分了工作臂和姿态协调臂, 自然应该用 \mathbf{J}_{AR}^k 。如果多个臂对等, 应尽量使用 \mathbf{J}_{AR}^k , 除非对每一个机械臂都同时给定了工作轨迹, 此时不存在多臂协调问题, \mathbf{J}_{AR}^k 也就求不出来了。

6.1.3 分解运动速度控制

姿态受限雅可比矩阵在速度水平上描述了第 k 个机械臂的末端效应器的运动速度和关节角速度的关系, 因此同样可用于分解运动速度控制。 \mathbf{J}_{AR}^{*k} 的应用同广义雅可比矩阵 \mathbf{J}^{*k} 一样。此处以 \mathbf{J}_{AR}^k 为例, 记 \mathbf{v}_d^k 是所期望的机械臂末端效应器的速度, 根据公式(6-14)有:

$$\dot{\boldsymbol{\phi}}_m^k = [\mathbf{J}_{AR}^k]^+ \mathbf{v}_d^k \quad (6-16)$$

将 $\dot{\boldsymbol{\phi}}_m^k$ 代入式(6-13), 求得 $\dot{\boldsymbol{\phi}}_m^{1-k}$ 后, 即可得 $\dot{\boldsymbol{\phi}}_m$, 它就是对本体不产生干扰的 FFSR 关节角速度控制向量。如果同时对多个机械臂的末端效应器给定的轨迹和速度进行跟踪, 则按照 \mathbf{J}_{AR}^k 的求解方法, 同理可得多个机械臂的基于多臂协调 ARGJM。在这种运动控制方法中, 要考虑以下因素:

(1) 选择合适的控制周期。分解运动速度控制方法通过在每个控制周期内输出机械臂各个关节的角速度实现对机械臂的运动控制。如果控制周期选得太大, 则运动控制的精度难以达到要求; 如果控制周期选得太小, 在每个周期内都要计算 \mathbf{J}_{AR}^k 的伪逆, 则给实时控制造成困难。

(2) 当 $\mathbf{v}_d^k \notin R(\mathbf{J}_{AR}^k)$ 时, 式(6-16)所求得的是最小二乘解, 机械臂处于奇异状态, 此时可以考虑选择该解, 或者选择上一个控制周期得到的 $\dot{\boldsymbol{\phi}}_m^k$ 控制机械臂运动。

(3) 应考虑机械臂关节角度和关节角速度的限制。

6.1.4 动力学奇异点与回避

利用广义雅可比矩阵和姿态受限广义雅可比矩阵按照给定的末端效应器的始末位置或在惯性空间中的给定路径求解关节角速度时,当广义雅可比矩阵 J^{*k} 和姿态受限广义雅可比矩阵 J_{AR}^k 奇异,则系统出现奇异点,无法得到逆问题解。由于FFSR系统的奇点与系统的惯性参数即质量几何特性有关,故称动力学奇异点。而地面机器人只具有运动学奇异点。另外FFSR满足的动量守恒定律一般情况下不可积分,故 J^{*k} 和 J_{AR}^k 不仅和 ϕ 的当前值有关,还和 ϕ 的运动路径和初始状态有关,机械臂的末端效应器的惯性空间坐标与 ϕ 不存在一一对应关系,具有非完整约束特性。所以动力学奇异点在数学上精确讨论求解十分困难。通常采取以下解决办法。

(1) 无冗余自由度 FFSR 系统。动力学奇点的本质是 J^{*k} 或 J_{AR}^k 矩阵不满秩,在物理上表现为无论机械臂各关节角如何运动都无法使机械臂末端效应器沿某一方向运动。为此,有两种方法:在奇异点附近对设计路径做微小改变以达到控制可实现性,对于路径的偏离可以采用反馈控制方法在 FFSR 系统通过奇异点领域后逐渐消除;启动本体的姿态控制系统,可消除该奇异点。

(2) 冗余自由度 FFSR 系统。机械臂处于奇异状态时,FFSR 系统至少失去一个自由度,为使机械臂末端严格跟踪给定的路径,应增加机械臂的自由度。

(3) 李雅普诺夫方法。对于给定的机械臂末端的位置而对运动路径不做具体要求时,可运用李雅普诺夫方法实现避免奇异点的路径规划。设 P_{nf} 为机械臂末端的期望的位置值,设计转角 $\dot{\phi}_m$ 的运动规律为:

$$\dot{\phi}_m^* = [J_{AR}^k]^+ A_V (P_{nf}^k - P_n^k) \quad (6-17)$$

$$\dot{\phi}_m = \begin{cases} \dot{\phi}_m^* & |\dot{\phi}_m^*| \leq \dot{\phi}_{max} \\ \text{sign}(\dot{\phi}_m^*) \dot{\phi}_{max} & |\dot{\phi}_m^*| > \dot{\phi}_{max} \end{cases} \quad (6-18)$$

式中, A_V 为正定常数矩阵。由于李雅普诺夫规划方法的渐进稳定性,此方法可保证最终趋向目标值。

6.2 基于 ARGJM 的 FFSR 运动规划

利用 ARGJM 规划 FFSR 的手臂运动捕捉目标,采用机械臂 1 先捕捉,然后用另一个机械臂捕捉的策略。

6.2.1 自由浮游状态下捕捉目标运动规划

在自由浮游状态下捕捉目标的运动规划,要求 FFSR 的机械臂能够准确抓住目标,同时维持机器人本体的姿态不变,为此,考虑到在捕捉目标之前,自由浮游机器人的惯性参数是已知的,因此采用了基于姿态受限的 FFSR 分解运动速度控制方法,而路径规划则采用李雅普诺夫方法,以避开奇异点。用机械臂 1 捕捉目标的路径规划算法如下。

算法 6-1 自由浮游机器人机械臂 1 捕捉目标路径规划算法

步骤 1 输入机器人的初始状态信息。根据机器人的几何参数和物理参数,计算机械

臂 1 的工作空间,设手把 1 的位置为 P_t , 机器人本体位置和姿态分别为 $r_0(0)$ 和 $\phi_s(0)$, $\phi_m(0)$ 为关节角向量。

步骤 2 若 P_t 不在工作空间之内,则无法捕捉,转步骤 11。

步骤 3 调节机器人的本体姿态,使 P_t 进入捕捉范围,置 $i = 0$ 。

步骤 4 计算机械臂 1 末端效应器的位置 $P_n^1(i)$ 。

步骤 5 若 $P_n^1(i) = P_t$,则捕捉成功,转步骤 11。

步骤 6 延着路径,进行运动速度分解,求得下一步的 v_d^1 。例如可令 $v_d^1 = (P_t - P_n^1(i))/(n \cdot \Delta t)$,若 v_d^1 太大,可增加进一步细分路径,如分为 n 份, Δt 为控制周期。

步骤 7 求 $\dot{\phi}_m^1(i) = (J_{FAR}^1)^+ v_d^1$,若 $|\dot{\phi}_m^1(i)| > \dot{\phi}_{max}^1$,则 $\dot{\phi}_m^1(i) = \text{sign}(\dot{\phi}_m^1(i)) \cdot |\dot{\phi}_{max}^1|$, 式中 $\dot{\phi}_{max}^1$ 为机械臂的最大关节运动速度。

步骤 8 利用式(6-13)求解相应的 $\dot{\phi}_m^2(i)$ 。按照 $\dot{\phi}_m(i)$ 驱动机械臂各关节运动。

步骤 9 计算机器人的新状态:

(1) 计算机器人本体的新位置 $r_0(i+1) = r_0(i) + \Delta r_0$ 。

(2) 计算机器人本体的姿态角 $\phi_s(i+1) = \phi_s(i) + \Delta \phi_s$ 。

(3) 计算机器人各关节的关节角 $\phi_m(i+1) = \phi_m(i) + \dot{\phi}_m(i) \Delta t$ 。

(4) 计算右臂末端效应器新的位置 $P_n^1(i+1)$ 。

步骤 10 $i = i + 1$,转步骤 5。

步骤 11 算法结束。

上述算法中应注意:步骤 7 中,当 $v_d^1 \in R(J_{FAR}^1)$ 时所求得的是最小二乘解,机械臂处于奇异状态,可以考虑选择该解,或者选择上一个控制周期得到的 $\dot{\phi}_m$ 控制机械臂运动,此时,本体姿态产生变化。

机器人的机械臂 1 捕捉目标之后,机械臂 1 的最后一个连杆和目标可看做为一个整体,即形成一个新的连杆。因此机械臂 1 最后一个连杆的长度、质量、转动惯量等参数均发生变化,运动控制更加复杂。

6.2.2 自由飞行状态下捕捉目标运动规划

在自由飞行状态下捕捉目标,因为扩大了 FFSR 的工作空间,受到最广泛的应用。当然此时仅仅控制机器人本体的位置和姿态,就可以使得机械臂的末端效应器跟踪给定的轨迹,这将消耗大量的燃料。为此,Spofford 和 Akin 提出了在自由飞行和自由浮游两状态间进行切换的控制方式,以减少 FFSR 在进行空间作业时对不可再生的燃料的消耗,收到很好的效果。因此,针对 FFSR 的运动控制亦采用该方法,即 FFSR 首先自由飞行,接近目标,使该目标进入自由浮游状态时的工作空间,关闭喷气装置和反作用轮,使系统进入自由浮游工作状态进行空间作业。

1. 自由飞行工作状态

自由飞行的工作任务就是使 FFSR 在适当的时间内以适当的运动方向和速度接近目标。主要问题包括:

(1) 会合点的选择。指的是 FFSR 通过自由飞行到达的、使操作目标处于其自由浮游

时的工作空间内的位置。确定会合点的前提是已知如下数据：目标的位置，FFSR 系统的位置和姿态，作为机器人本体的卫星的飞行动力学参数等。会合点的选择可以按不同的标准来确定。

(2) 轨迹设计。要考虑运动轨迹的可行性、安全性。根据目标和 FFSR 是否在一个轨道上，决定 FFSR 的轨迹规划问题，如是否需要变轨控制等。关于这方面的具体研究请参见有关卫星轨道控制的文献。

2. 自由浮游工作状态

FFSR 进入自由浮游工作状态后，即可采用调运动规划算法。需要注意的一点是由于 FFSR 的自由飞行，使得 FFSR 本体的质量等惯性参数发生变化，因此，需要进行实时的在线辨识。综上所述，提出如下算法：

算法 6-2 FFSR 捕捉目标运动规划算法

步骤 1 输入 FFSR 和目标的初始位姿信息。

步骤 2 根据机器人的几何参数和物理参数，计算机械臂 1 的工作空间，设目标上的手把 1 的位置为 P_t ，机器人本体位置和姿态分别为 $r_o(0)$ 和 $\phi_s(0)$ ， $\phi_m(0)$ 为关节角向量。

步骤 3 若 P_t 在工作空间之内，转步骤 7。

步骤 4 选择会合点，规划空间机器人的飞行轨迹。

步骤 5 合拢双臂，启动喷气装置，接近会合点。

步骤 6 到达会合点后，关闭喷气设备和姿态控制系统，进入自由浮游状态，双臂对称运动，恢复初始最佳构性。

步骤 7 利用参数辨识算法进行机器人本体参数辨识。

步骤 8 利用新参数重新计算姿态受限广义雅可比矩阵。

步骤 9 利用算法 6-1 规划机械臂运动，捕捉目标，进行空间作业。

步骤 10 算法结束。

上述算法中，步骤 5 和步骤 6 中的机械臂合拢和恢复也是 FFSR 从自由飞行到进入自由浮游状态的关键步骤之一。机械臂合拢的主要目的：一是减小系统的转动惯量，有利于用喷气装置驱动 FFSR 的自由飞行；二是避免 FFSR 与太空中其他物体的碰撞而造成损坏。合拢和恢复均在自由浮游状态下进行，同样可以利用姿态受限广义雅可比矩阵进行规划，以减少对机器人本体的姿态干扰。

6.2.3 计算机仿真

为验证提出运动规划算法的可行性及效率，以双臂 6 自由度 FFSR 模型为例，进行了计算机仿真。

1. 自由浮游状态

首先采用基于姿态受限广义雅可比矩阵的自由浮游状态下的运动规划算法控制机器人的机械臂 1 沿直线运动，其仿真结果如图 6-1 所示。在机械臂的末端完成给定任务的同时，另一个臂进行协调运动，本体的姿态并不发生变化，仅仅是产生了平移，如图 6-1 中的本体中心轨迹是垂直移动的。

其次分别进行了三个仿真实验，并针对实验结果进行了比较。针对如 FFSR 和目标的模型，在相同初始姿态的情况下，完成捕捉目标的运动规划。图 6-2 和图 6-3 给出了该仿真

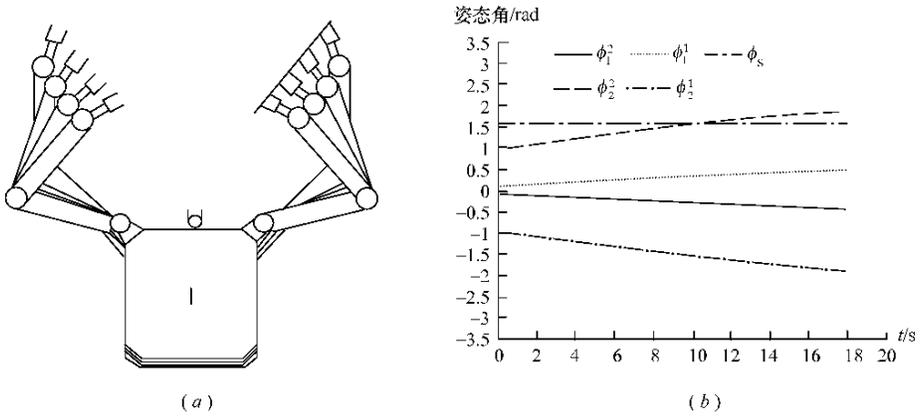


图 6-1 FFSR 双臂协调运动仿真结果

(a) 机械臂 1 沿直线运动,机械臂 2 协调; (b) 本体姿态角和机械臂关节角变化轨迹。

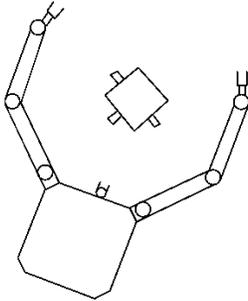


图 6-2 自由浮游的初始状态

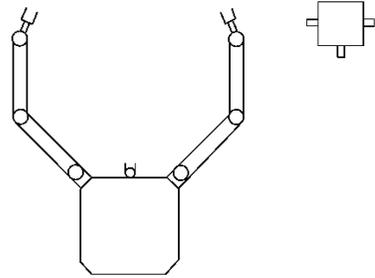


图 6-3 自由飞行的初始状态

实验的两种初始状态。

(1) 仿真对比实验一。不采用任何姿态控制算法控制 FFSR 捕捉目标,即利用 Umentani 提出的广义雅可比矩阵进行运动规划,其仿真结果如图 6-4 所示。其中图(a)为 FFSR 捕捉目标的运动过程,图(b)和图(c)所示分别为运动过程中本体中心变化轨迹和本体姿态角的变化轨迹。从仿真结果中可以看出在机械臂的运动过程中,尽管机器人捕捉到目标,但本体的中心位置和姿态均产生了很大的变化。

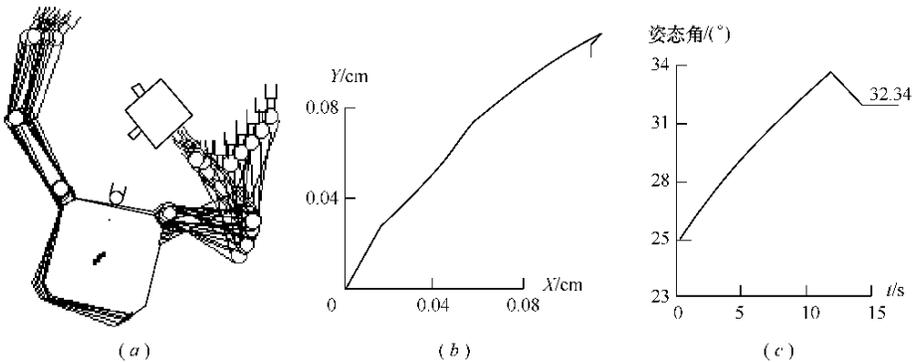


图 6-4 在无姿态控制状态下捕捉目标的仿真结果

(a) 捕捉目标; (b) 本体中心变化轨迹; (c) 本体姿态角变化轨迹。

(2) 仿真对比实验二。采用 S. Dubowsky 提出的基于增强干扰图(EDM)的姿态控制算法控制 FFSR 捕捉目标。增强干扰图是描述空间机械臂对本体干扰的一种方法,它通过在关节空间规划机械臂的运动来减少到机器人本体的姿态干扰。首先在机械臂关节空间内选择一些点,计算对飞行器本体的动力学干扰最小的运动方向,沿干扰最小的方向连成曲线,就构成干扰图。在干扰图上用不同颜色的点或不同分布的点表示最大干扰值的变化,就形成了增强干扰图。从关节空间中某一点出发沿干扰最小的方向运动形成一条运动轨迹,沿着该轨迹运动将对本体的姿态干扰最小。图 6-5 和图 6-6 分别给出了图 2-2 所示的 DFBSR 右臂的增强干扰图和受限最小干扰图。

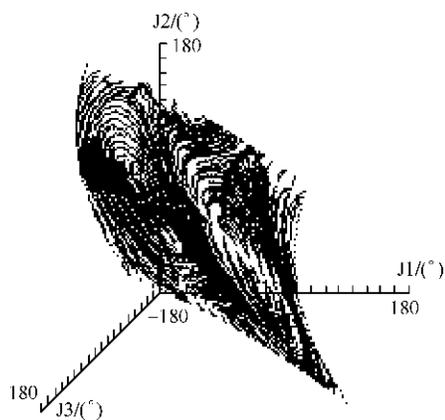


图 6-5 增强干扰图

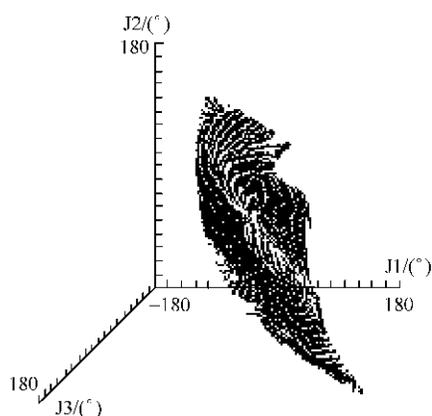


图 6-6 受限最小干扰图

利用增强干扰图进行运动规划的仿真结果如图 6-7 所示。其中图(a)为 FFSR 机械臂捕捉目标的运动过程,图(b)和图(c)所示分别为运动过程中本体中心变化轨迹和本体姿态角的变化轨迹。从仿真结果中可以看出在机械臂的运动过程中,尽管机器人捕捉到目标,本体的中心位置和姿态均发生了的变化,但姿态变化要比无姿态控制时的情况要小得多。

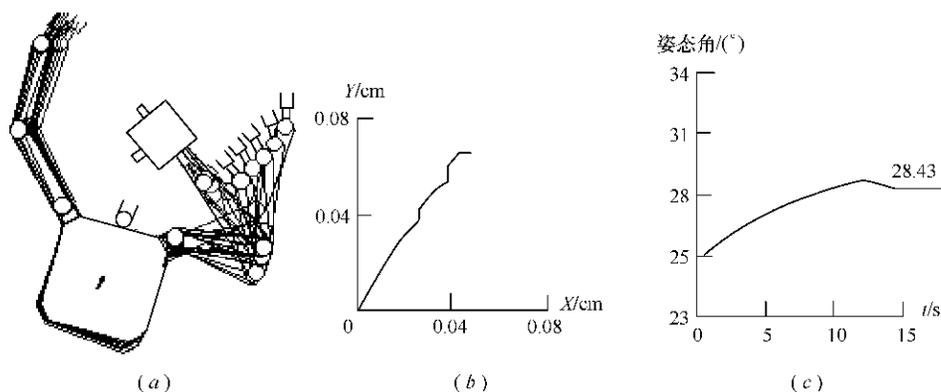


图 6-7 利用 S. Dubowsky 提出的姿态控制算法捕捉目标的仿真结果

(a) 捕捉目标; (b) 本体中心变化轨迹; (c) 本体姿态角变化轨迹。

(3) 仿真对比实验三。采用提出的基于姿态受限广义雅可比矩阵的运动控制算法,规划 FFSR 机械臂的协调运动以捕捉目标,使其完成空间作业任务的同时,不对 FFSR 的本体姿态产生干扰,其仿真结果如图 6-8 所示。其中图(a)为 FFSR 机械臂捕捉目标的运动过

程,图(b)和图(c)所示分别为运动过程中本体中心变化轨迹和本体姿态角的变化轨迹。从仿真结果中可以看出在机械臂的运动过程中,尽管机器人的本体位置发生漂移,但其姿态却保持不变。

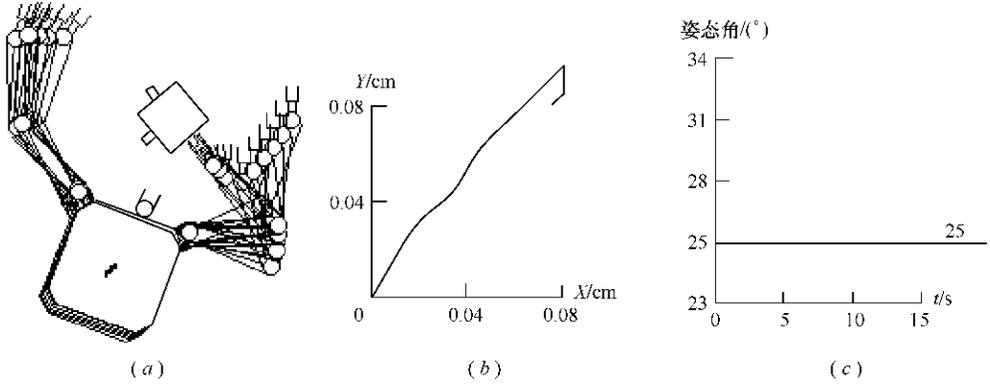


图 6-8 使用基于 ARGJM 的姿态控制方法捕捉目标的仿真结果

(a) 捕捉目标; (b) 本体中心变化轨迹; (c) 本体姿态角变化轨迹。

在上面三个计算机仿真实验中,可以看出,实验一中的姿态干扰最大,实验二中的姿态干扰较小,而实验三中无姿态变化。因此不需消耗能量来使反作用轮或反作用喷气装置工作,以维持本体姿态不变,从而节省能量,延长 FFSR 的轨道寿命。另外本章的方法同最小干扰图方法相比,有三点优势:

(1) 不需要计算干扰图,从而节省大量的存储空间。

(2) 针对多臂冗余系统,干扰图的计算量随指数上升,本章的算法是线性增加。

(3) 对于有空间障碍物的环境,关节空间由于被障碍物分割,不再是一整块空间,EDM 所给的路径很可能不是避碰的,而本章算法则不受限制。

因此从仿真结果比较中,可以得到如下结论:基于 ARGJM 的 FFSR 运动控制算法计算速度快,占用内存空间小,姿态稳定效果也较好。

2. 自由飞行状态

当目标位于自由浮游机器人的工作空间之外时,利用本章的策略,FFSR 必须首先飞行到目标附近,然后进入自由浮游状态,完成空间作业。本仿真内容包括以下三个内容(图 6-3 给出了该仿真实验的初始状态)。

(1) 不对机械臂进行控制,仅仅利用喷气装置控制本体的位置和姿态,进而控制机械臂末端效应器的位置和姿态,从而捕捉目标。仿真结果如图 6-9 所示。

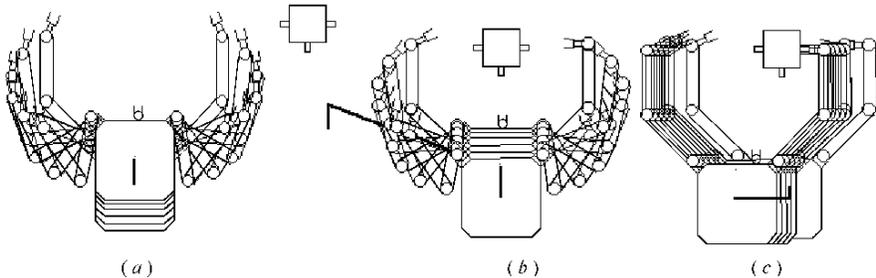


图 6-9 利用喷气装置控制本体捕捉目标

(2) FFSR 飞行到目标附近, 然后进入自由浮游状态, 不利用参数辨识, 而是利用 W. Walker 提出的自适应控制算法驱动 FFSR 进行空间作业。仿真结果如图 6-10 所示。

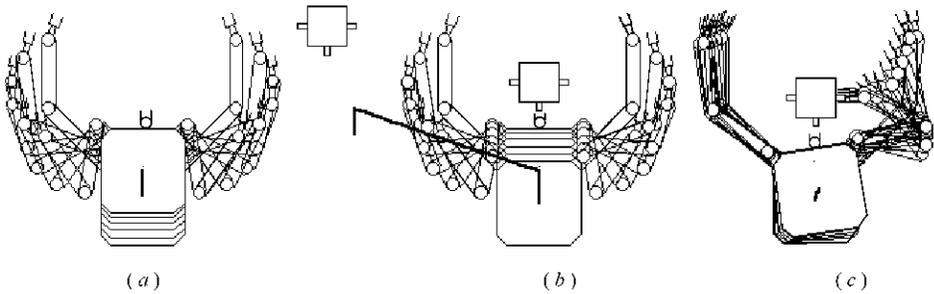


图 6-10 利用自适应控制算法捕捉目标

(3) FFSR 飞行到目标附近, 然后进入自由浮游状态, 利用本章提出的参数辨识方法和基于 ARGJM 运动规划方法进行空间作业。仿真结果如图 6-11 所示。

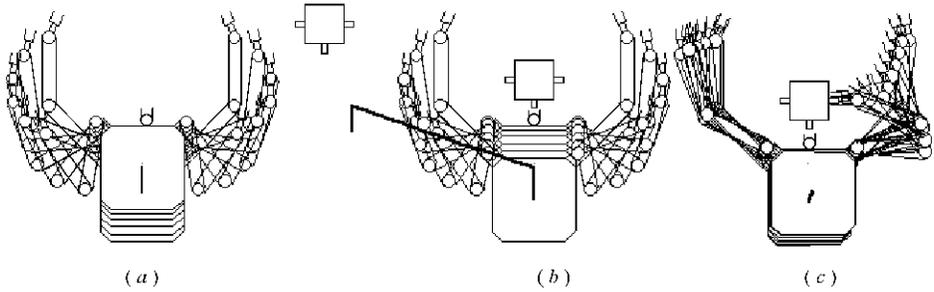


图 6-11 利用本章算法捕捉目标

从仿真结果可以看出, 三种算法都可以抓住目标, 但是图 6-9 仅仅采用喷气装置控制本体的位置和姿态进而抓住目标, 不需要参数辨识, 很明显它消耗了大量的燃料, 并且增加了喷气装置驱动控制系统的设计难度。图 6-10 采用自适应控制的方法, 控制机械臂运动捕捉目标, 但是该方法不能进行多臂协调运动规划, 消除对本体的姿态干扰。图 6-11 为本文提出的算法, 首先辨识本体的惯性参数, 其次利用基于姿态受限广义雅可比矩阵的多臂协调运动规划算法规划机械臂的运动捕捉目标, 同时对本体的姿态干扰为零。从仿真结果可以看出无论在燃料消耗还是在姿态控制的效果方面, 所提出的算法都比实验中的另两种算法要好。

6.3 小 结

为了安全和更有效地规划多个机械臂和本体之间的协调运动, 本章提出了多臂冗余 FFSR 的姿态受限广义雅可比矩阵的概念, 分析了机械臂动力学奇异点的回避方法, 并给出了利用姿态受限广义雅可比矩阵完成基于姿态干扰最小的多臂 FFSR 的运动规划和运动控制算法, 最后针对自由浮游和自由飞行两种状态进行了计算机仿真和实验比较, 证明所提出算法的可行性和有效性。

第七章 FFSR 姿态控制综合算法

对于冗余 FFSR,理论上可以通过协调运动规划消除对本体的姿态干扰,但其应用误差亦需要利用姿态控制系统来消除;另外对本体进行姿态控制还可以解决 FFSR 的动力学奇异点回避问题,特别是抓住目标后系统的惯性参数产生变化,奇异点更易发生。

对于非冗余 FFSR 系统,机械臂运动必然会对卫星本体的姿态产生干扰,必须利用卫星的姿态控制系统以维持姿态的稳定,因此应该进一步研究该情况下如何尽量减少姿态控制的能量消耗。

7.1 卫星的姿态控制

FFSR 的本体既是卫星,必须建立和保持一定的姿态,因此需要进行姿态控制。姿态控制系统是由姿态敏感器、姿态控制和卫星的动力学组成的闭环控制系统。姿态敏感器能直接测量某个选定的参考天体或目标在形体坐标系中的方向或与该方向有关的信息,经过比较和运算得出卫星在空间参考坐标系中的状态。姿态控制器包括控制信号处理和力矩执行机构,喷气、磁控属于外力矩控制,飞轮电机实现内力矩控制。而利用各种飞轮与星体之间的角动量交换是姿态稳定控制的主要模式。图 7-1 为卫星的姿态控制框图。

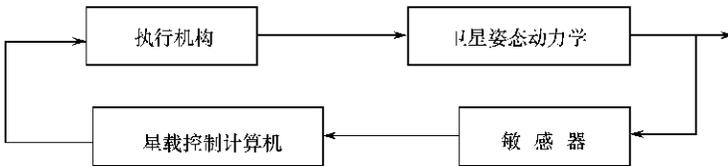


图 7-1 卫星的姿态控制框图

根据动力学特性,三轴姿态控制系统分为两类。① 零动量系统。系统内部动量矩为零或很小,因而不具有陀螺定轴性。它的特点是当姿态误差很小时,三个姿态轴的控制是独立的。② 偏置动量系统。内部具有陀螺定轴性的高速飞轮。其特点是三轴间互相耦合和具有较慢的响应速度(扰动量矩方向除外)。本书以零动量系统为例。由于姿态角偏差较小时,三个方向的姿态运动不耦合,因此,其姿态控制方程可写为:

$$\begin{aligned} I_x \ddot{\phi} &= -\dot{h}_x + T_{cx} + T_{dx} \\ I_y \ddot{\theta} &= -\dot{h}_y + T_{cy} + T_{dy} \\ I_z \ddot{\psi} &= -\dot{h}_z + T_{cz} + T_{dz} \end{aligned} \quad (7-1)$$

式中, I_x 、 I_y 和 I_z 为卫星绕相应轴的转动惯量; T_c 为喷气控制力矩; T_d 为外部干扰力矩; h 对应与星体内部角动量的变化,此处指的是反作用飞轮角动量的变化量。考虑到单纯喷气

控制姿态的缺点,目前多采用三个正交安装的飞轮进行姿态控制。它们的角动量与星体的主惯量轴平行,分别独立地吸收沿滚动、俯仰、偏航轴的外绕角动量。姿态控制系统根据姿态敏感器测的姿态偏差,调节飞轮的转速以实现连续式的姿态控制。以偏航通道为例,根据动力学方程式(7-1),有偏航轴姿态单纯利用反作用轮的控制方程:

$$I_z \ddot{\psi} = -\dot{h}_z + T_{dz} = J_z \dot{\omega} + T_{dz} \quad (7-2)$$

式中, $J_z \dot{\omega}$ 为反作用轮的转动惯量和转速的乘积。考虑到飞轮电机的运动特性和姿态敏感器的测量特性,引入 PID 控制器,可得偏航通道姿态控制回路的传递函数框图,如图 7-2 所示。

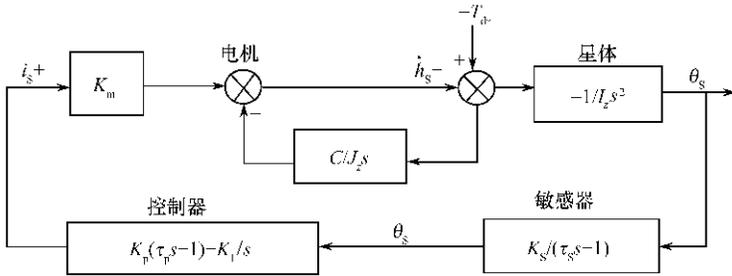


图 7-2 偏航通道姿态控制回路框图

在图 7-2 中, τ_s 为姿态敏感器的时间常数; K_m 为电机的力矩系数; C 为飞轮轴承的摩擦因数。略去 τ_s , 该控制系统的闭环传递函数为:

$$\frac{\theta_s}{T_{dz}} = \frac{J_z s + c}{I_z s^2 (J_z s + c) + J_z [K_p (\tau_p s + 1) s + K_I] K_m K_s} \quad (7-3)$$

其稳态误差为:

$$\theta_{ss} = \frac{c T_{dz}}{J_z K_1 K_m K_s} \quad (7-4)$$

若飞轮轴承的摩擦因数 $C = 0$, 显然, 系统的稳态误差亦为零。由式(7-2)可知, 稳态时飞轮的转速变化为:

$$\dot{\omega} = T_{dz} / J_z \quad (7-5)$$

当飞轮的转速超过容许范围的上界(或下界)时, 必须沿偏航轴的负(或正)方向施加一个去饱和用的喷气力矩 T_{cz} , 在图 7-2 中的 T_{dz} 被 T_{cz} 代替, 同理, 该控制系统将保持姿态稳定, 并使飞轮的转速按相反方向回到初值, 即将飞轮储存的外扰动量释放出去, 使飞轮可以重新吸收外扰动量, 这就是飞轮去饱和的过程。如果干扰力矩大于反作用飞轮调节的极限, 将直接利用喷气装置控制卫星的姿态。由此可见, 在空间机器人的机械臂运动过程中, 应尽量减少对本体的干扰, 以免飞轮饱和, 从而消耗喷气装置使用的燃料。

7.2 基于姿态干扰预测的姿态控制算法

7.2.1 基本思想

目前, 许多文献研究 FFSR 的运动控制及如何减少对本体姿态干扰的运动规划问题, 而很少研究实际应用中由于 FFSR 的几何参数和惯性参数不精确等原因产生的姿态稳定问题, 认为这可以单纯利用上节提出的卫星姿态控制系统进行校正。这种方法确实可行,

但是该方法没有充分利用机械臂运动时的预测信息,会消耗大量的姿态控制能量。

为此,研究基于姿态干扰预测的姿态控制方法。由于现在的卫星姿态控制系统非常成熟,因此可以作为一个整体模块进行利用。而在机械臂运动之前,将估计其对本体姿态产生干扰的反作用力,并传递给姿态控制系统以完成前馈控制补偿,从而提高姿态控制系统的整体性能。估计反作用力的形式可以是反作用力矩,也可以是角动量。因为姿态控制系统在进行姿态控制时可利用反作用轮产生该控制力矩或控制角动量。

然而,从事先规划的 FFSR 机械臂的运动轨迹估计反作用力矩是比较困难的,因为规划产生的仅仅是关节的轨迹和角速度,而不是关节驱动力矩。机械臂的力矩控制一般情况仅仅应用于力接触时或抓取目标时被采用。而沿着规划的路径和轨迹估计角动量是容易的,所生产的误差可以由姿态控制系统的增益裕度来吸收。

卫星姿态控制系统(SAC)利用姿态敏感器测量的数据完成闭环控制。当机械臂运动时,SAC 将利用机械臂运动控制系统提供的反作用角动量的估计值进行前馈控制,同时利用姿态敏感器的输出进行反馈控制,共同维持本体的姿态稳定。而姿态控制的状态如卫星的姿态、运动速率和反作用轮的控制状态可以提供给机械臂运动控制系统修正其运动规划。

综上所述,基于姿态干扰预测的姿态控制系统的框图如图 7-3 所示。

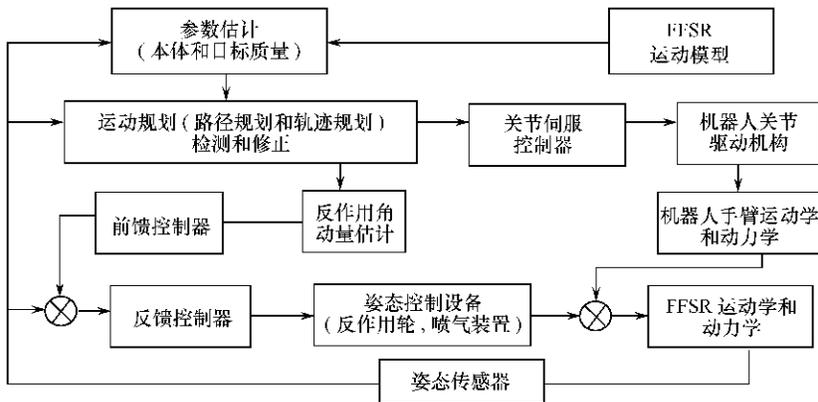


图 7-3 基于姿态干扰预测的复合姿态控制框图

7.2.2 理论分析

反馈控制系统的实质是在对象受到干扰后,必须在被控量出现偏差时,才按照系统输出偏差的大小产生作用以补偿干扰对被控量的影响,所以反馈控制本身决定了无法将干扰克服在被控量偏离设定值之前,从而限制了控制质量的进一步提高。

然而通过在控制系统中引入前馈控制,即直接按扰动进行控制,从而在被控量还未显示出变化之前,控制器就产生了作用,从而克服反馈控制的缺点。因此前馈和反馈控制相结合,就有可能既减少系统的稳态误差,又保证系统稳定。从图 7-3 提取姿态控制系统可得如图 7-4 所示的基于姿态干扰预测的前馈反馈复合控制传递函数框图。其中 $G_1(s)$ 是反馈控制器的传递函数, $G_2(s)$ 是卫星的反作用轮或喷气系统的传递函数, $G_3(s)$ 对应 FFSR 的动力学传递函数。 $N(s)$ 为干扰力矩或角动量。由于输入 $R(s)$ 为零,则输出 $C(s)$ 为:

$$C(s) = \frac{[1 - F_n(s)G_2(s)]G_3(s)}{1 + G_1(s)G_2(s)G_3(s)}N(s) \tag{7-6}$$

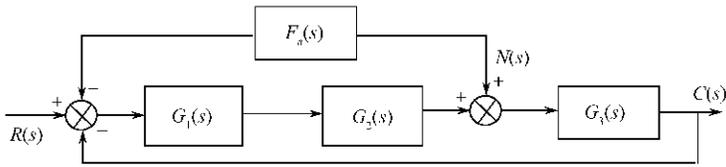


图 7-4 基于姿态干扰预测的复合控制传递函数框图

如果选择 $F(s) = 1/(G_1(s)G_2(s))$ ，则输出 $C(s)$ 完全不受扰动的影响，系统的暂态和稳态误差都为零。因为该方法是通过零点、极点对消实现的，并要求 $F_n(s)$ 是理想的微分环节，另外还存在元件的参量和性能的变化，补偿愈加困难。因此实际应用时只能得到近似解。详细分析请参考有关的控制理论。

关于对本体的反角动量预测问题，Longman 等人提出一种介于牛顿 - 欧拉动力学方程和拉格朗日方程之间的方法，计算比较复杂。考虑到动量守恒定律的实际意义，计算机械臂的动量增量，即为它对本体的反作用角动量，如式(7-7) 所示。

$$\Delta L = \sum_{k=1}^l \sum_{i=1}^{n_k} (I_i^k \omega_i^k + m_i^k r_i^k)(t = t_{i+1}) - \sum_{k=1}^l \sum_{i=1}^{n_k} (I_i^k \omega_i^k + m_i^k r_i^k \times r_i^k)(t = t_i) \quad (7-7)$$

7.2.3 计算机仿真

针对图 2-2 的 FFSR 模型，对基于姿态干扰预测的姿态控制算法进行了仿真。针对偏航轴，机械臂的运动会对本体的姿态产生变化。比较在独立的反作用轮控制和基于姿态干扰预测的姿态控制情况下，系统的动态性能和能量消耗的比较。仿真框图如图 7-5 所示。为了简化仿真模型，做了如下假设：①在仿真过程中，忽略环境对本体的姿态干扰；②忽略机器人上的柔性附件；③姿态传感器具有足够的精度和足够快的响应时间。

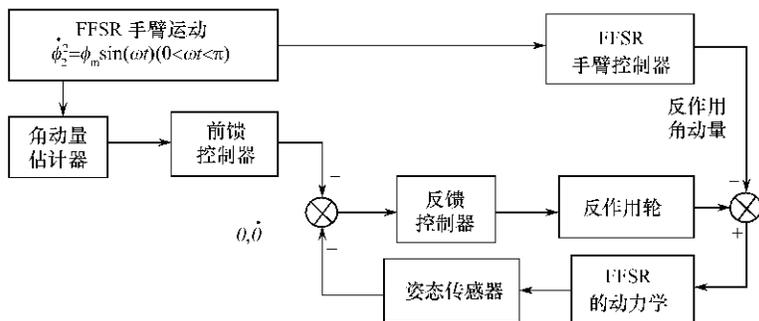


图 7-5 动态响应仿真框图

设定 FFSR 的第 2 个机械臂的第 2 个关节角 ϕ_2^2 给定正弦输入，分别进行如下三个仿真。

(1) 姿态控制系统不运行。在这种情况下，关闭卫星的姿态控制系统，FFSR 工作在自由浮游状态，显示出典型的姿态干扰特征。该情况在捕捉巨大的浮游目标是经常用到，当空间任务完成后，必须立即启动姿态控制系统，矫正空间机器人本体的姿态，以继续保持通信连接和太阳能帆板对太阳的定向。计算机仿真如图 7-6(a) 和图 7-7(a) 所示。

(2) 单独采用反作用轮姿态控制。在这种情况下，机械臂运动时对本体产生了姿态干扰，反作用姿态控制系统立即启动，实时矫正空间机器人本体的姿态，以始终保持通信连接和太阳能帆板对太阳的定向。计算机仿真如图 7-6(b) 和图 7-7(b)、图 7-8(a) 所示。

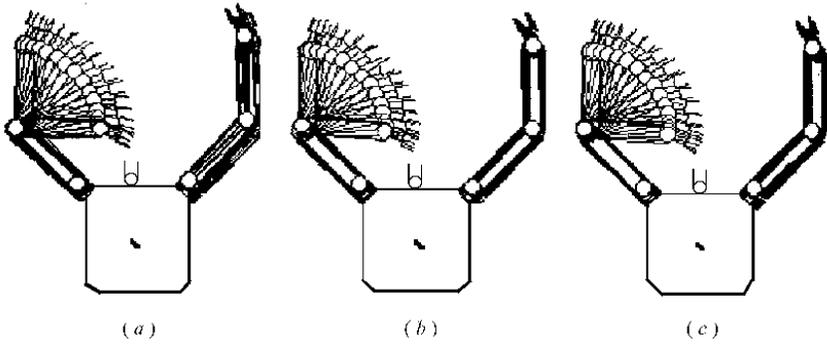


图 7-6 姿态控制系统动态响应过程

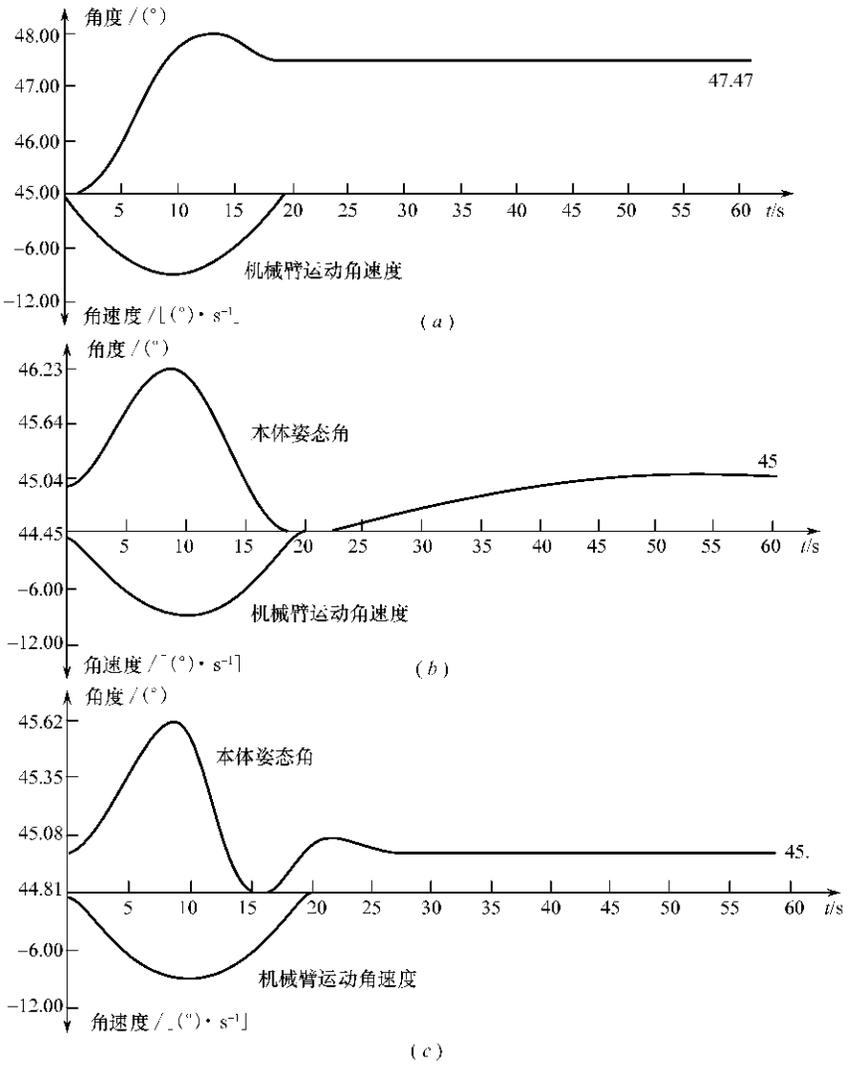


图 7-7 不同姿态控制方式下姿态角变化曲线

(a) 无姿态控制; (b) 反作用轮控制; (c) 基于姿态干扰预测的复合姿态控制方法。

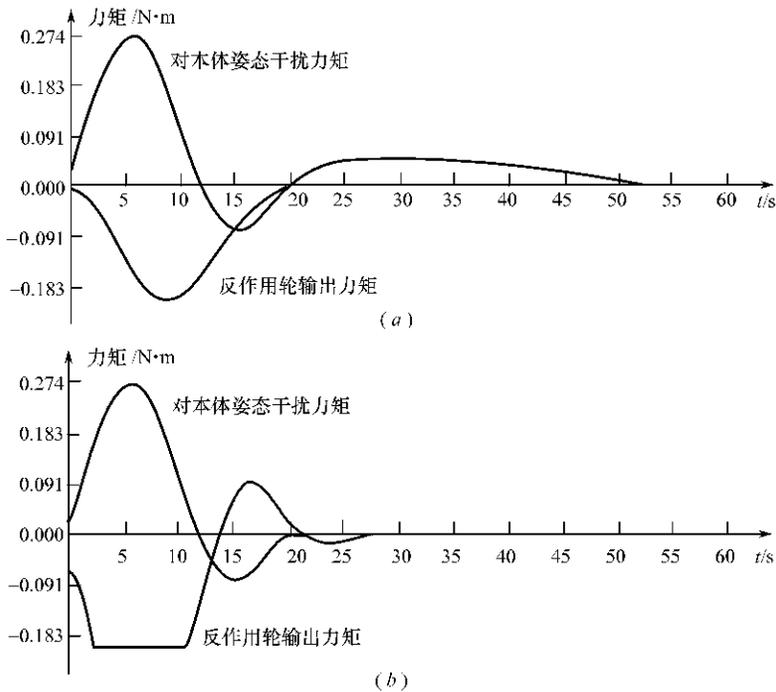


图 7-8 力矩变化曲线

(3) 基于姿态干扰预测的复合姿态控制算法。在这种情况下,机械臂运动时对本体产生了姿态干扰,反作用姿态控制系统立即启动,利用预测的反作用角动量进行前馈补偿,同时利用姿态敏感器的输出进行反馈控制,实时矫正 FFSR 本体的姿态。计算机仿真如图 7-6(c) 和图 7-8(b) 所示。

从仿真结果可以看出,采用基于姿态干扰预测的复合姿态控制方案,同独立的反作用轮控制方法比较,可以有效减少姿态干扰,提高系统的响应性能。

7.3 FFSR 复合姿态控制算法

7.3.1 复合姿态控制算法

FFSR 进行空间作业,大体可以分为两阶段:捕捉到目标以前的运动控制和捕捉到目标以后的运动控制。由于两阶段系统的惯性参数和几何结构发生变化,因此应该针对不同阶段提出相应的控制策略,从而形成统一的 FFSR 进行空间作业的自主协调复合姿态控制算法,以达到每一阶段基于姿态控制消耗最小的最优控制。对于机械臂末端执行器在接触到目标直到捕捉到目标的过程中复杂的力控制,即柔顺捕捉过程,本书不做具体介绍,请参阅相关文献。

相对而言,捕捉到目标以前的运动控制算法比较简单。因为此时 FFSR 机械臂的惯性参数是已知的,而本体的惯性参数可以利用本书提出的基于神经网络的参数辨识算法精确地估计出来。即 FFSR 的广义雅可比矩阵(GJM)或姿态受限广义雅可比矩阵(ARGJM)所需的几何参数和惯性参数均可以较精确地获得。而这些参数在设计时均经过综合考虑,

谨慎选择,以保证 FFSR 的工作空间和避免在最佳的工作范围内出现动力学奇异点。对于冗余系统,一般情况下,也保证各参数能够满足机械臂和本体的协调运动控制。因此,这一阶段,可以利用本书提出的基于参数辨识的 FFSR 多臂协调运动规划算法,在实际应用中的由于参数误差和实际驱动机械臂运动时的关节角速度误差而产生的姿态干扰,可以单纯利用反作用轮姿态控制系统进行矫正,以减少控制的复杂性。

由于捕捉到目标以后,可以认为目标和机械臂的末端结合为一整体,改变了系统惯性参数,其运动控制算法相对复杂。表现为:

(1) 另一个机械臂的运动干扰本体的姿态和位置,从而也干扰了目标的位置和姿态。

(2) 在许多情况下,目标的几何参数和惯性参数也是未知的。目标的几何参数相对可以较精确地测量,而惯性参数的在线辨识则比较困难,下面将给出一个利用神经网络的辨识算法。即使对于目标的几何参数和惯性参数可以精确知道,由于其具有较大的变化范围,对空间机器人系统惯性参数的影响是较大的。此时得用协调运动规划算法时,便可能无解。即无论机械臂的运动如何协调运动,都无法避免对本体的姿态干扰。

(3) 由于需要多臂一起协调工作,无疑增加了机械臂的任务空间,而本体的自由度却不发生变化。就有可能由捕捉前的运动冗余系统变为非冗余系统,当然,此时机械臂运动必然将对本体的姿态产生干扰。

在上述(2)、(3)情况下,便可以利用基于姿态干扰预测的姿态控制算法,控制本体的姿态,以提高系统的性能。当然,如果捕捉到目标以后,系统的惯性参数分布仍然能够满足机械臂对本体协调的运动规划,就没有必要采用复合姿态控制了。综上所述,该算法的过程如图 7-9 所示。

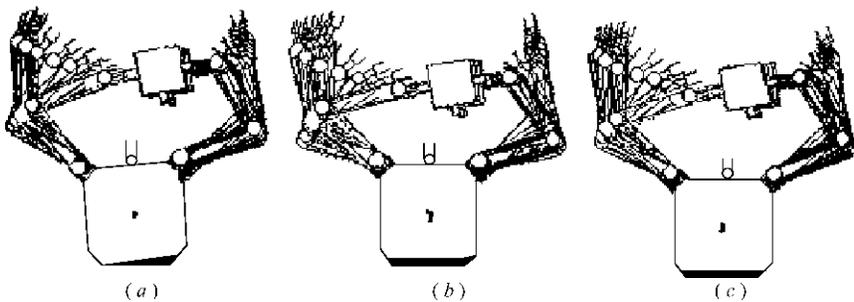


图 7-9 空间作业过程

以多臂 FFSR 完成空间作业为例的姿态控制综合算法如下。其中,先用一个机械臂捕捉目标,之后根据作业要求,规划其他机械臂的运动。

算法 7-1 自主协调运动控制算法

步骤 1 利用算法 6-2 控制 FFSR 的某一机械臂捕捉目标。

步骤 2 若目标的惯性参数已知,转步骤 3,否则,在线辨识目标的惯性参数。

步骤 3 判断其他机械臂是否完成任务。若成功,转步骤 5。

步骤 4 根据系统对机械臂作业的要求,判断系统在新的任务空间 m_2 变化的情况下是否冗余。不冗余转步骤 6。

步骤 5 利用 ARGJM 规划机械臂的运动。利用单纯姿态控制系统控制实际工作中的姿态误差。

步骤 6 利用姿态受控广义雅可比矩阵(ACGJM)规划机械臂的运动,估计其对本体的

干扰角动量,驱动机械臂运动并启动基于姿态干扰预测的姿态控制系统运行。转步骤 3。

步骤 7 算法结束。

上述算法的步骤 6 中,由于启动姿态控制设备,FFSR 系统不再满足角动量守恒,本书定义此时代表机械臂末端运动速度和关节角速度关系的矩阵为姿态受控广义雅可比矩阵 ACGJM (Attitude-Controlled GJM)。

利用反作用轮或喷气装置控制本体的姿态,使其维持不变,而本体可以自由浮动。则由式(2-69)可得:

$$\mathbf{v}^k = \mathbf{J}_s \dot{\boldsymbol{\phi}}_s + \mathbf{J}_m^k \dot{\boldsymbol{\phi}}_m \quad (7-8)$$

由于姿态控制系统的作用,可认为 $\dot{\boldsymbol{\phi}}_s = 0$,代入式(6-14)可得:

$$\mathbf{v}^k = \mathbf{J}_m^k \dot{\boldsymbol{\phi}}_m \quad (7-9)$$

式(7-9)中的 \mathbf{J}_m^k 称为姿态受控广义雅可比矩阵(ACGJM)。它与姿态受限广义雅可比矩阵(ARGJM)是不同的,可比较式(6-8)和式(6-15)。两者的主要区别是:ACGJM 所描述的机械臂的运动对本体产生干扰,但由于启动卫星的姿态控制系统,本体的姿态不发生变化;而 ARGJM 所描述的机械臂的运动理论上就不对本体产生姿态干扰,因此本体的姿态不发生变化。

7.3.2 捕捉目标后的空间作业运动规划

以图 2-2 提出的 FFSR 模型为例,针对双臂 FFSR 完成空间作业进行分析。即先用机械臂 1 捕捉目标,然后用机械臂 2 捕捉目标。

第六章中,已经详细地论述了用机械臂 1 捕捉目标运动算法和控制策略。本节重点分析用机械臂 2 捕捉目标的算法。机器人的机械臂 1 捕捉目标之后,主要有两个特点:

(1) 机械臂 1 的最后一个连杆和目标可看做为一个整体,即形成一个新的连杆。因此机械臂 1 最后一个连杆的长度、质量、转动惯量等参数发生变化,必然先利用参数辨识算法,获得这些参数,然后重新计算系统质心和姿态受限广义雅可比矩阵。

(2) 机械人的机械臂 2 运动时,会使机器人本体的位置和姿态改变,从而使目标的位置和姿态改变。因此机械臂 2 捕捉目标的运动规划算法关键在于基于视觉反馈的实时规划。

另外如果捕捉到目标以后,系统的惯性参数分布和机器人的自由度相对于新的机械臂的任务空间而言是冗余的。既然能够满足机械臂和本体的协调运动规划,就没有必要基于姿态干扰预测的复合姿态控制,只需利用反馈校正应用误差即可。此处设定捕捉到目标后,捕捉到目标的那一个机械臂的各个关节锁定,系统变为非冗余系统,从而可验证本书提出的基于姿态干扰预测的复合姿态控制的优越性。

算法 7-2 FFSR 机械臂 2 捕捉目标的运动规划算法

步骤 1 根据机器人的几何参数、物理参数和机器人左臂捕捉目标后的状态,设机器人本体位置为 $\mathbf{P}_0(0)$,本体姿态为 $\boldsymbol{\phi}_0(0)$, $\boldsymbol{\phi}_M(0)$ 为关节角向量,目标上的手把 2 的位置为 $\mathbf{P}_1(0)$ 。

步骤 2 置 $i = 0$ 。计算机械臂 2 末端执行器的位置 $\mathbf{P}_n^2(i)$ 。

步骤 3 计算 $\Delta \mathbf{P} = |\mathbf{P}_n^2(i) - \mathbf{P}_1(i)|$,若 $\Delta \mathbf{P} < \Delta \mathbf{P}_d$,则捕捉成功,转步骤 11。

步骤 4 辨识机械臂 1 末端执行器的几何参数和惯性参数。

步骤 5 计算 $v_d^2 = \Delta P / \Delta t$, Δt 为控制周期。

步骤 6 计算 $\dot{\phi}_M(i) = J^{*2} v_d^2$, 若 $|\dot{\phi}_M(i)| > \dot{\phi}_{\max}$, 则 $\dot{\phi}_M(i) = |\dot{\phi}_{\max}|$, 式中 $\dot{\phi}_{\max}$ 为机械臂最大关节运动速度。

步骤 7 根据式(6-12) 估计对本体的干扰力矩。

步骤 8 按照 $\dot{\phi}_M(i)$ 驱动机械臂各关节运动, 同时启动基于姿态干扰预测的姿态控制系统运行。

步骤 9 计算机器人的新状态:

- (1) 计算机器人本体的新位置 $P_0(i+1) = P_0(i) + \Delta P_0$ 。
- (2) 利用姿态敏感器测量此时空间机器人本体姿态角 $\phi_0(i+1)$ 。
- (3) 计算机器人各关节的关节角 $\phi_M(i+1) = \phi_M(i) + \dot{\phi}_M(i)\Delta t$ 。
- (4) 计算机械臂 2 末端执行器新的位置 $P_c^r(i+1)$ 。
- (5) 计算目标的新状态 $P_t(i+1)$ 。

步骤 10 转步骤 4。

步骤 11 算法结束。

为验证提出的运动规划算法的可行性及效率, 本节以图 2-2 提出的 FFSR 模型为例, 针对双臂 FFSR 完成空间作业进行了计算机仿真。

具体包括三个内容: ① 捕捉目标前后均采用无姿态控制的 GJM 方法; ② 捕捉目标前采用基本姿态受限广义雅可比方法, 捕捉目标后, 采用单纯的反作用轮姿态控制方法。③ 捕捉目标前采用基于姿态受限广义雅可比方法, 捕捉目标后, 采用基于姿态干扰预测的复合姿态控制方法。仿真结果如图 7-9 和图 7-10 所示。

图 7-9(a)、(b)、(c) 和图 7-10(a)、(b)、(c) 分别对应无姿态控制、单纯采用反作用轮姿态控制和基于姿态干扰预测的复合姿态控制的 FFSR 捕捉目标的运动过程和本体姿态变化曲线。可以看出, 由于采用了姿态控制, FFSR 在进行空间作业时, 后两种情况下的本体姿态最终保持稳定。基于姿态干扰预测的复合姿态控制的动态过程比单纯采用反作用轮姿态控制时间短, 本体姿态变化范围明显见效, 从图 7-11 可以看出, 采用基于姿态干扰预测的复合姿态时维持姿态控制时的能量消耗也明显减小, 从而取得了较好的姿态控制效果。

7.4 小 结

本章提出了 FFSR 的姿态控制综合算法, 它将 FFSR 的机械臂运动控制系统和本体的姿态控制系统相结合, 其中机械臂运动控制系统负责运动规划以尽量减少对本体的姿态干扰并提供干扰力矩预测, 姿态控制系统完成基于姿态干扰预测的复合姿态控制。该方案比上述两个系统独立工作有效地减少了姿态控制误差和燃料消耗, 较好地解决了冗余 FFSR 姿态控制的应用误差和避免奇异点问题, 以及非冗余 FFSR 的姿态控制问题。最后, 针对 FFSR 完成捕捉目标的不同工作过程, 利用该算法进行了计算机仿真, 验证了其可行性。

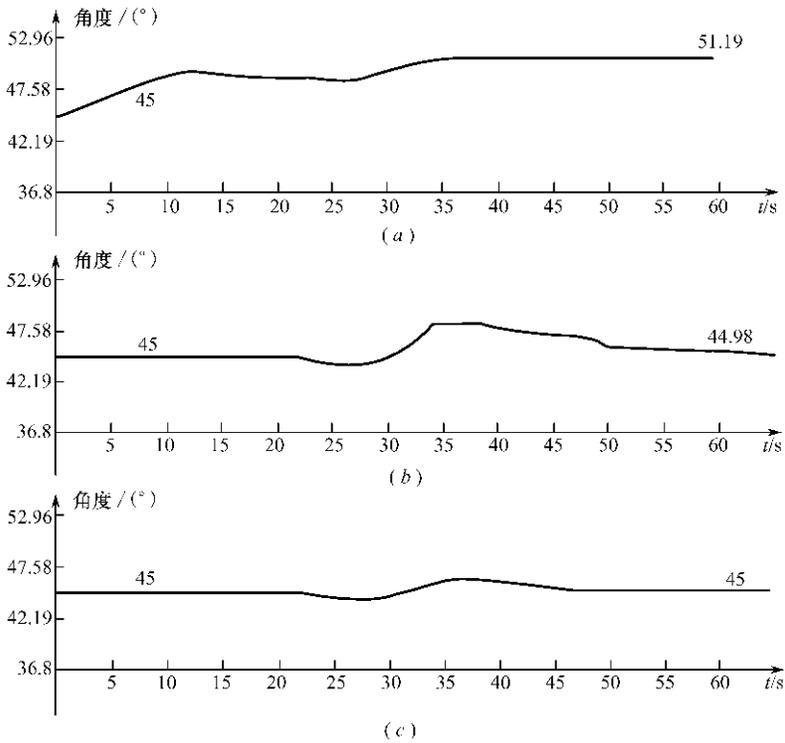


图 7-10 本体姿态变化曲线

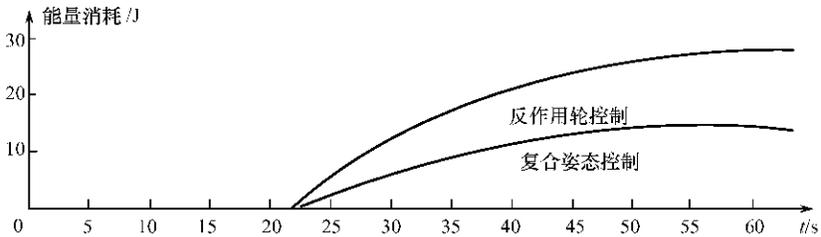


图 7-11 本体姿态控制能量耗比较

第八章 基于关节力矩的 FFSR 运动控制算法

FFSR 关节力矩的有效递推算算法是遥控和自主控制中的一项基本技术。近年来，FFSR 的结构设计和控制研究是 FFSR 领域研究和关注的焦点。FFSR 可以代替航天员完成各种舱外活动等复杂作业，克服由航天员探索空间面临的险恶、危险环境。但由于空间特有的微重力环境及 FFSR 没有固定基座，它不能直接应用传统地面机器人的基于关节驱动力矩的控制方法。因此，研究 FFSR 在微重力下的基于关节驱动力矩的运动控制方法

是十分必要的。

8.1 关节驱动力矩递推算法

机器人各关节力矩的有效计算方法,不仅在对机器人机械结构和控制器进行设计时,而且在进行基于力矩的实时控制时,都起着十分重要的作用。特别地,从力矩控制的角度来看,为确保控制的实时性、可靠性和稳定性,需要高效的力矩计算算法。Luh 等人在这方面做了大量研究工作,针对地面机器人提出了递推 N-E 法,它是以理论力学的两个基本方程——牛顿方程和欧拉方程为出发点,结合机械手的速度和加速度分析而得出的一种有效的机械手动力学算法。该法是由固定基座前推,即向末端递推,逐次求出各杆的角速度、角加速度和质心加速度,再由末杆的末关节向第一关节(与基座相连)后推,从而求出各关节的力矩。但由于 FFSR 的本体不固定,该法不适用于 FFSR,而将 Luh 的递推 N-E 法推广到 FFSR 的关键是如何有效地求出卫星本体的速度和加速度。

为此,根据 Luh 等人提出的地面机器人的 N-E 法,以 FFSR 两相邻连杆之间的基本运动学和动力学关系为出发点,将机械手各连杆的运动分解成两部分,一部分与机器人本体运动有关,另一部分与机器人本体运动无关,而可由递推算法决定。并考虑 FFSR 系统固有的运动特性——动量守恒定律和力平衡法则,提出了一种用于计算 FFSR 关节驱动力矩的有效递推算法。为验证本书提出关节力矩算法的有效性,做了计算机仿真。下面给出计算 FFSR 机械手各连杆速度、加速度及关节驱动力矩的有关递推公式。

8.1.1 运动模型

本章所研究的 FFSR 几何模型如图 8-1 所示。这里做如下假设:

- (1) 整个系统由 $n+1$ 个刚性连杆构成,通过 n 个转动关节连接,每个关节有一个转动自由度,由力矩控制器控制;
- (2) 没有任何外力 / 力矩作用在系统上,在 FFSR 的操作过程中,整个系统满足线动量和角动量守恒定理;
- (3) 假设系统的初始动量为零,并忽略摩擦和微重力影响;

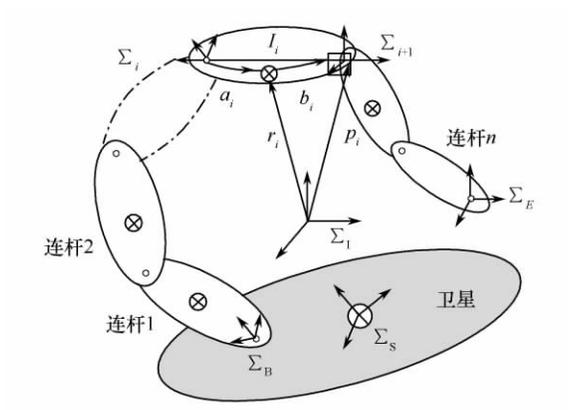


图 8-1 FFSR 几何模型

(4) 卫星本体的位置和姿态不受其他装置控制。

本章采用基本符号约定如下：

Σ_1 : 固定于系统质心的惯性坐标系；

Σ_S : 固定在本体质心 C_0 的坐标系；

Σ_B : 机械手基座坐标系；

Σ_i : 固定于关节 i 的连杆坐标系, 其 z 轴与关节 i 轴一致; 第 i 个连杆坐标系, 对 $i = 1, 2, \dots, n$, 分别对应机械手的第 i 个连杆坐标系, 第 n 个坐标系对应机械手末端效应器坐标系, $i = 0$ 对应本体坐标系；

${}^j R_i \in R^{3 \times 3}$: 从坐标系 Σ_i 到 Σ_j 的变换矩阵；

${}^1 R_B \in R^{3 \times 3}$: 从惯性坐标系到机械手基座坐标系的旋转矩阵；

${}^1 R_S \in R^{3 \times 3}$: 卫星本体相对于 Σ_1 的姿态矩阵；

${}^1 R_i \in R^{3 \times 3}$: 从惯性坐标系到第 i 个连杆的坐标系, $i = 0$ 对应本体坐标系, $i = 1, 2, \dots, n$ 对应机械手第 i 个连杆坐标系；

$z_i \in R^3$: 关节轴的方向, 与 Σ_i 系的 z 轴方向一致；

n : 机械手的自由度；

m_i : 第 i 个连杆的质量(kg), 第 0 个连杆是卫星本体；

m_C : FFSR 系统的总质量；

$l_i \in R^3$: 从 Σ_i 原点到 Σ_{i+1} 原点的位置向量或连杆 i 的长度(m)；

${}^i I_i \in R^{3 \times 3}$: 在第 i 个连杆坐标系中表示的第 i 个连杆相对其质心的惯性矩阵 ($\text{kg} \cdot \text{m}^2$)；

${}^1 I_i \in R^{3 \times 3}$: 在惯性坐标系中表示的第 i 个连杆相对其质心的惯性矩阵 ($\text{kg} \cdot \text{m}^2$)；

${}^1 r_i \in R^3$: 在惯性坐标系中表示的从惯性坐标系原点到第 i 个连杆质心的位置向量 (m)；

${}^B r_i \in R^3$: 在机械手基座坐标系 Σ_B 中表示的从机械手基座坐标系原点到第 i 个连杆质心的位置向量 (m)；

$p_i \in R^3$: 关节 i 的位置向量；

$a_i \in R^3$: 从关节 $i - 1$ 指向连杆 i 质心的位置向量；

$b_i \in R^3$: 从关节 i 质心指向连杆 i 的位置向量；

$\theta_i, \dot{\theta}_i, \ddot{\theta}_i$: 关节 i 的角度、角速度和角加速度；

${}^j \dot{r}_i, {}^j \ddot{r}_i$: 坐标系 Σ_j 中连杆 i 的质心的速度和加速度；

${}^j \dot{p}_i, {}^j \ddot{p}_i$: 坐标系 Σ_j 中关节 i 的速度和加速度；

${}^j \omega_i, {}^j \epsilon_i$: 坐标系 Σ_j 中连杆 i 的速度和加速度；

${}^j f_i, {}^j n_i$: 坐标系 Σ_j 中作用在连杆 i 的质心的力和力矩；

τ_i : 连杆关节 i 的驱动力矩；

${}^1 z_i \in R^3$: 在惯性坐标系中表示的沿第 i 个连杆坐标系 z 轴方向的单位向量 (m)；

${}^1 \omega_i \in R^3$: 在惯性坐标系中表示的沿第 i 个连杆的角速度 (rad/s)；

$\theta_M \in R^n$: 机械手关节角向量 ($\theta_M = [\theta_1 \cdots \theta_n]^T$) (rad)

$J_M^i \in R^{6 \times n}$: 在惯性坐标系中表示的第 i 个连杆质心的位姿雅可比矩阵 ($i = 1, 2, \dots, n$);

$E_i \in R^{i \times i}$: $i \times i$ 单位矩阵;

α, β, γ : 为用 $z-y-x$ 欧拉角 (rad) 描述的卫星本体的姿态角, 记 $\theta_0 = \theta_s = [\alpha \beta \gamma]^T$ 。

向量 $r_i = [r_{ix}, r_{iy}, r_{iz}]^T$ 的算符 \hat{r}_i 代表如下的斜对称矩阵操作:

$$\hat{r}_i = \begin{bmatrix} 0 & -r_{iz} & r_{iy} \\ r_{iz} & 0 & -r_{ix} \\ -r_{iy} & r_{ix} & 0 \end{bmatrix}$$

在符号左边的上标表示向量描述的坐标系。当向量在惯性坐标系 Σ_1 表示时, 该上标通常被省略。

8.1.2 FFSR 的速度分析

FFSR 的速度分析指根据给定的机械手所有关节的角度和角速度来求解 FFSR 的各个连杆的速度和角速度。应用刚体运动学基本定理描述相邻两连杆之间运动学关系, 各连杆的角速度、关节速度和质心速度可表示成如下递推式:

$${}^i\omega_i = {}^iR_{i-1}(\dot{\theta}_i z_i + {}^{i-1}\omega_{i-1}) \quad (8-1)$$

$${}^i\dot{p}_i = {}^i\omega_i \times {}^{i-1}l_i + {}^iR_{i-1}{}^{i-1}\dot{p}_{i-1} \quad (8-2)$$

$${}^i\dot{r}_i = {}^i\omega_i \times b_i + {}^i\dot{p}_i \quad (8-3)$$

机器人本体角速度和速度在坐标系 Σ_0 可表示成:

$${}^0\omega_0 = R_0^T \omega_0 \quad (8-4)$$

$$\dot{p}_0 = R_0^T (v_0 + \omega_0 \times p_0) \quad (8-5)$$

根据式(8-1)和式(8-4), 将连杆角速度 ${}^i\omega_i$ 分解成两部分, 一部分与本体的角速度 ω_0 有关, 另一部分与本体的角速度 ω_0 无关而可用递推算法获得; 利用归纳法可以得到如下计算 FFSR 各连杆角速度的简单递推式:

$${}^i\omega_i = {}^i\bar{\omega}_i + R_i^T \omega_0 \quad (8-6)$$

式中, ${}^i\bar{\omega}_i$ 满足下面的递推关系:

$${}^0\bar{\omega}_0 = 0, {}^i\bar{\omega}_i = {}^iR_{i-1}(\dot{\theta}_i z_i + {}^{i-1}\bar{\omega}_{i-1}) \quad (8-7)$$

类似地, 根据式(8-2), 可将连杆关节速度 ${}^i\dot{p}_i$ 分解成两部分, 一部分与卫星的角速度 ω_0 和线速度 v_0 有关, 另一部分可用递推算法获得; ${}^i\dot{p}_i$ 可简化成下面的递推式:

$${}^i\dot{p}_i = {}^i\bar{p}_i + R_i^T (v_0 + \omega_0 \times p_0) \quad (8-8)$$

式中

$${}^0\bar{p}_0 = 0, {}^i\bar{p}_i = {}^i\bar{\omega}_i \times {}^{i-1}l_i + {}^iR_{i-1}{}^{i-1}\bar{p}_{i-1} \quad (8-9)$$

同样地, 根据式(8-3)各连杆质心速度 ${}^i\dot{r}_i$ 可表示成如下简单递推式:

$${}^i\dot{r}_i = {}^i\bar{r}_i + R_i^T (v_0 + \omega_0 \times r_i) \quad (8-10)$$

式中

$${}^i\bar{r}_i = {}^i\bar{\omega}_i \times b_i + {}^i\bar{p}_i \quad (8-11)$$

从以上的速度和角速度分析可知,只要先求出机器人本体的速度和角速度,即可用递推法获得 FFSR 机械手各连杆的速度和角速度。

8.1.3 FFSR 的加速度分析

FFSR 的加速度分析指根据给定的机械手所有关节的角度、角速度和角加速度来求解 FFSR 的各个连杆的角加速度和加速度。对速度向量式(8-1)、式(8-2)和式(8-3)求导,并考虑向量在两相对转动坐标系 Σ_i 和 Σ_{i+1} 的导数关系式可得各连杆的角加速度、关节加速度和质心加速度的递推关系式:

$${}^i\dot{\omega}_i = {}^iR_{i-1}(\ddot{\theta}_i z_i + {}^{i-1}\omega_{i-1} \times \dot{\theta}_i z_i + {}^{i-1}\omega_{i-1}) \quad (8-12)$$

$${}^i\ddot{p}_i = {}^i\dot{\omega}_i \times {}^{i-1}l_i + {}^i\omega_i \times ({}^i\omega_i \times {}^{i-1}l_i) + {}^iR_{i-1}{}^{i-1}\ddot{p}_{i-1} \quad (8-13)$$

$${}^i\ddot{r}_i = {}^i\dot{\omega}_i \times {}^i b_i + {}^i\omega_i \times ({}^i\omega_i \times {}^i b_i) + {}^i\ddot{p}_i \quad (8-14)$$

设 FFSR 本体的加速度和角加速度在惯性坐标系 Σ_1 表示为 α_0 和 ϵ_0 ,类似地,可将各连杆的角加速度分解成两部分,一部分与本体的角加速度 ϵ_0 有关,另一部分与本体的角加速度 ϵ_0 无关而可用递推算法获得;由式(8-12)可得各连杆角加速度的递推式:

$${}^i\epsilon_i = {}^i\dot{\omega}_i = {}^i\bar{\epsilon}_i + R_i^T \epsilon_0 \quad (8-15)$$

式中

$${}^0\bar{\epsilon}_0 = 0, {}^i\bar{\epsilon}_i = {}^iR_{i-1}(\ddot{\theta}_i z_i + {}^{i-1}\omega_{i-1} \times \dot{\theta}_i z_i + {}^{i-1}\bar{\epsilon}_{i-1}) \quad (8-16)$$

同样地,由式(8-13)可得各连杆关节的加速度的递推式:

$${}^i\alpha_i = {}^i\ddot{p}_i = {}^i\bar{\alpha}_i + R_i^T(\alpha_0 + \epsilon_0 \times p_i) \quad (8-17)$$

式中

$$\begin{cases} {}^0\bar{\alpha}_0 = R_0^T(\omega_0 \times v_0 + \omega_0 \times \omega_0 \times p_0) \\ {}^i\bar{\alpha}_i = {}^i\bar{\epsilon}_i \times {}^{i-1}l_i + {}^i\omega_i \times ({}^i\omega_i \times {}^{i-1}l_i) + {}^iR_{i-1}{}^{i-1}\bar{\alpha}_{i-1} \end{cases} \quad (8-18)$$

应该注意的是:在坐标系 Σ_0 观察到的 FFSR 本体的加速度等于从惯性坐标系 Σ_1 观察到的本体加速度与哥氏加速度和离心加速度的向量和。

由式(8-14)可得各连杆质心的加速度递推式:

$${}^i\alpha_{ci} = {}^i\ddot{r}_i = {}^i\bar{\alpha}_{ci} + R_i^T(\alpha_0 + \epsilon_0 \times r_i) \quad (8-19)$$

式中

$$\begin{cases} {}^0\bar{\alpha}_{c0} = R_0^T[\omega_0 \times (\omega_0 \times b_0)] + {}^0\bar{\alpha}_0 \\ {}^i\bar{\alpha}_{ci} = {}^i\bar{\epsilon}_i \times {}^i b_i + {}^i\omega_i \times ({}^i\omega_i \times {}^i b_i) + {}^i\bar{\alpha}_i \end{cases} \quad (8-20)$$

从以上的角加速度和加速度分析可知,只要先求出 FFSR 本体的角加速度和加速度,即可用递推法获得 FFSR 机械手各连杆的角加速度、关节的加速度和质心的加速度。

8.1.4 FFSR 的关节驱动力矩计算公式

根据牛顿定律和欧拉角动量方程可获得作用在连杆 i 上的力和力矩:

$${}^i f_i = m_i \alpha_{ci}, {}^i n_i = {}^i I_i \epsilon_i + {}^i \omega_i \times {}^i I_i \omega_i \quad (8-21)$$

将式(8-15)、式(8-17)和式(8-19)代入式(8-21)可得:

$${}^i f_i = {}^i \bar{f}_i + m_i R_i^T (\alpha_0 + \varepsilon_0 \times r_i), {}^i n_i = {}^i \bar{n}_i + I_i R_i^T \varepsilon_0 \quad (8-22)$$

式中

$${}^i \bar{f}_i = m_i {}^i \bar{\alpha}_{c_i}, {}^i \bar{n}_i = I_i {}^i \bar{\varepsilon}_i + {}^i \bar{\omega}_i \times I_i {}^i \bar{\omega}_i \quad (8-23)$$

根据 N-E 法, 不考虑摩擦和重力影响时, 作用在各连杆上的关节驱动力矩可求得:

$$\tau_{f,i} = {}^i R_{i+1} \tau_{f,i+1} + {}^i f_i \quad (8-24)$$

$$\tau_{n,i} = {}^i R_{i+1} \tau_{n,i+1} + {}^{i-1} l_i \times ({}^i R_{i+1} \tau_{f,i+1}) + ({}^{i-1} l_i + {}^i b_i) \times {}^i f_i + {}^i n_i \quad (8-25)$$

$$\tau_i = \tau_{n,i} {}^i R_{i-1} z_i \quad (8-26)$$

以上各连杆的速度、加速度和力矩计算公式表明一旦求出本体的速度、角速度、加速度和角加速度, 即可获得 FFSR 各关节的驱动力矩。因此, 将 Luh 的递推 N-E 法推广到 FFSR 的关键是如何有效求出机器人本体的速度和加速度。FFSR 在不受外力和外力矩作用下, 整个系统满足动量和角动量守恒定律; 假定初始动量和角动量为零可得:

$$\sum_{i=0}^n m_i \dot{r}_i = 0, \sum_{i=0}^n (I_i \dot{\omega}_i + m_i r_i \times \dot{r}_i) = 0 \quad (8-27)$$

由力和力矩平衡原理可得:

$$\sum_{i=0}^n {}^1 f_i = 0, \sum_{i=0}^n ({}^1 n_i + r_i \times {}^1 f_i) = 0 \quad (8-28)$$

而将上面各连杆的速度和角速度递推式代入式(8-27), 即可计算出机器人本体的速度 v_0 、角速度 ω_0 :

$$v_0 = M_1^{-1} L_v, \omega_0 = M_2^{-1} L_\omega \quad (8-29)$$

式中

$$M_1 = \text{diag}(m_c, m_c, m_c), M_2 = \sum_{i=0}^n (R_i I_i R_i^T - m_i \hat{r}_i \hat{r}_i^T) \quad (8-30)$$

$$L_v = - \sum_{i=0}^n R_i m_i \dot{r}_i, L_\omega = - \sum_{i=0}^n [R_i I_i \dot{\omega}_i + r_i \times (R_i m_i \dot{r}_i)] \quad (8-31)$$

根据以上求得的 FFSR 本体的速度和角速度, 由式(8-6)、式(8-7)和式(8-8)即可获得 FFSR 的各个连杆的速度和角速度。

将式(8-22)代入式(8-28)可求得卫星本体的加速度 α_0 和角加速度 ε_0 :

$$\alpha_0 = M_1^{-1} F_\alpha, \varepsilon_0 = M_2^{-1} F_\gamma \quad (8-32)$$

式中, 矩阵 M_1 和 M_2 与式(8-30)相同。

$$F_\alpha = - \sum_{i=0}^n R_i {}^i \bar{f}_i, F_\gamma = - \sum_{i=0}^n [R_i {}^i \bar{n}_i + r_i \times (R_i {}^i \bar{f}_i)] \quad (8-33)$$

至此类似 N-E 法, 可将 FFSR 的关节力矩递推算法表示成两部分: ① 速度、加速度和惯性力前向递推; ② 约束力和关节力矩后向递推。其递推算法计算步骤描述如下。

算法 8-1 FFSR 关节驱动力矩递推算法

步骤 1 预先给定 FFSR 的连杆几何和物理参数, 如连杆长度 l_i 、质量 m_i 、惯量 I_i 和变量初值; 给定 FFSR 机械手各关节的 $\theta_i, \dot{\theta}_i, \ddot{\theta}_i (i = 1, \dots, n)$; 并离线计算 $m_c = \sum_{i=0}^n m_i$ 等值。

步骤 2 确定机械手各连杆之间的变换矩阵: $R_0, {}^0 R_1, \dots, {}^{i-1} R_i, \dots, {}^{n-1} R_n$; 用式 ${}^j R_i = {}^j R_{j+1} {}^{j+1} R_{j+2} \dots {}^{i-1} R_i$, 来计算 ${}^i R_i$ 。

步骤 3 根据式(8-7)、式(8-9)和式(8-11),计算 ${}^i\dot{\omega}_i$ 、 ${}^i\dot{p}_i$ 和 ${}^i\dot{r}_i$ 。

步骤 4 用式(8-30)~式(8-31)计算矩阵 M_1 、 M_2 、 L_v 和 L_ω 。

步骤 5 根据式(8-29)求解卫星本体的速度 v_0 和角速度 ω_0 。

步骤 6 用式(8-6)、式(8-8)和式(8-10),计算机械手各连杆的角速度 ${}^i\omega_i$ 、速度 ${}^i\dot{p}_i$ 和其质心速度 ${}^i\dot{r}_i$ 。

步骤 7 根据式(8-16)、式(8-18)和式(8-20),计算各连杆中间值 ${}^i\bar{\epsilon}_i$ 、 ${}^i\bar{\alpha}_i$ 和 ${}^i\bar{\alpha}_{ci}$ 。

步骤 8 用式(8-33)计算 F_α 、 F_γ 。

步骤 9 根据式(8-32)求解卫星本体的加速度 α_0 和角加速度 ϵ_0 。

步骤 10 用式(8-15)、式(8-17)和式(8-19),计算各连杆的角加速度 ${}^i\epsilon_i$ 、加速度 ${}^i\alpha_i$ 和质心的加速度 ${}^i\alpha_{ci}$ 。

步骤 11 用式(8-22)~式(8-23),计算作用在各连杆上的力 if_i 和力矩 in_i 。

步骤 12 根据式(8-24)~式(8-26),计算各关节的驱动力矩 τ_i 。

用算法 8-1 计算关节驱动力矩,对 n DOF 旋转关节的 FFSR 而言,所需计算量大约为:乘法为 $330 \times n + 190$ 次;加法为 $416 \times n + 280$ 次。总共需要为 $330 \times n + 190$ 次乘法和 $416 \times n + 280$ 次加法,即关节力矩递推算法的计算量数量级为 $O(n)$ 。换言之,关节力矩递推算法的计算量与 FFSR 的自由度成正比。而对同样具有 n DOF 旋转关节的地面机器人而言,计算关节驱动力矩所需的总计算量为 $133 \times n - 18$ 次乘法和 $106 \times n - 20$ 次加法。

8.2 基于关节力矩的捕捉目标控制算法

FFSR 控制问题主要表现在捕捉目标时的运动控制问题和随即而产生的姿态控制问题。FFSR 的控制方式主要分为如下四类。

(1) 采用装载在本体上的喷气装置来补偿机械手对本体位姿的干扰作用,保持本体位姿的稳定,从而可直接利用地面机器人的控制技术。

(2) 采用喷气装置或反作用飞轮仅控制本体的姿态而不控制其位置。

(3) 本体不受控,可在机械手运动影响下自由调整位姿。

(4) 采用推进器使 FFSR 达到期望的位置和姿态,完成工作空间无限制的作业任务。

上节的关节力矩计算是在假定 FFSR 的机械手各关节的运动 θ_i 、 $\dot{\theta}_i$ 、 $\ddot{\theta}_i$ 已知的条件下进行的,在实际应用中,应根据具体的空间作业任务规划出所需的 FFSR 机械手各关节的运动 θ_i 、 $\dot{\theta}_i$ 、 $\ddot{\theta}_i$ 。

本节基于上节所提出的关节驱动力矩求解算法并结合基于广义雅可比矩阵(GJM)的分解运动加速度控制(RMAC)和分解运动速度控制(RMRC)法,提出了一种捕捉目标的力矩控制算法。

8.2.1 控制策略

末端效应器速度与机械手关节速度之间的关系为:

$$\dot{x}_E = J_M^* \dot{\theta}_M \quad (8-34)$$

式(8-34)表明 FFSR 末端效应器的速度与关节角速度的关系,在形式上跟地面机器人的描述方法一样。应该特别指出,这里给出的 $GJM(J_M^*)$ 是在惯性坐标系定义的。而升谷等人给出的 $GJM({}^0J_M^*)$ 是在本体坐标系中定义的,两者相差一坐标变换 ${}^1A(\theta_0)$,即有关系:

$$J_M^*(\theta_0, \theta_M) = {}^1A(\theta_0) {}^0J_M^*(\theta_M) \quad (8-35)$$

针对机械手的基本控制,研究机械手末端效应器速度层次上的直角坐标空间控制方法,即通常所称的分解运动速度控制(RMRC),是一种基于机械手逆运动学控制方法。它将在直角坐标空间描述的末端效应器操作速度命令分解成关节运动命令来完成给定任务。控制策略可由雅可比矩阵的逆来定义。

下面给出 FFSR 在星载和惯性空间的分解运动速度控制(RMRC)的控制策略。

$$\text{在本体坐标系:} \quad \dot{\theta}_M = ({}^0J_M^*)^{-1} \mathbf{v}_d \quad (8-36)$$

$$\text{在惯性坐标系:} \quad \dot{\theta}_M = (J_M^*)^{-1} \mathbf{v}_d \quad (8-37)$$

式中, \mathbf{v}_d 是所期望的机械手末端效应器速度。在式(8-36)和式(8-37)中,雅可比矩阵的奇异点是不同的,前者是运动学奇异性,后者是动力学奇异性,然而两者都具有相同的形式,它表明基于地面固定基座机器人开发的控制策略可用于空间机械手,只需更换雅可比矩阵。

类似地,根据 H. L. Alexander 和 R. H. Cannon, Y. Umetani 和 K. Yoshida 提出的 FFSR 的分解运动加速度控制法可得:

$$\ddot{\theta}_M = J_M^* [\dot{\mathbf{v}}_d + k_v(\mathbf{v}_d - \dot{\mathbf{p}}_n) + k_p(\mathbf{p}_d - \mathbf{p}_n) - \dot{J}_M^* \dot{\theta}_M] \quad (8-38)$$

8.2.2 FFSR 捕捉目标控制算法

捕捉目标是 FFSR 在空间执行的一项重要作业任务。FFSR 要安全、可靠并高效地完成捕捉任务,应主要考虑以下几个方面的问题:① 能量消耗;② 力矩控制;③ 控制效率。综合考虑以上几方面因素,并结合基于 GJM 的 RMAC 控制,提出一种捕捉目标的力矩控制算法。该控制算法的步骤如下。

算法 8-2 基于关节驱动力矩的捕捉目标控制算法

步骤 1 预先给定 FFSR 的连杆几何和物理参数,如连杆长度 l_i 、质量 m_i 、惯量 I_i 和物理模型的 D-H 参数值及变量初值, $i = 1 \sim n$; 离线计算 $m_c = \sum_{i=0}^n m_i$ 等值;并输入 FFSR 的初始状态信息,根据 FFSR 的几何和物理参数,计算机械手的工作空间(WS)。设目标上手柄的位置为 \mathbf{p}_d ,机械手初始关节向量为 $\theta_M(0)$,卫星本体姿态角为 $\theta_0(0)$;控制周期为 Δt 。

步骤 2 若 \mathbf{p}_d 不在 WS 之内,则无法捕捉目标,转步骤 10。

步骤 3 调节 FFSR 的本体姿态,使 \mathbf{p}_d 进入捕捉范围,且置 $j = 0$ 并确定机械手各连杆之间的变换矩阵: $\mathbf{R}_0, {}^0\mathbf{R}_1, \dots, {}^{i-1}\mathbf{R}_i, \dots, {}^{n-1}\mathbf{R}_n$; 用公式 ${}^j\mathbf{R}_i = {}^j\mathbf{R}_{j+1} {}^{j+1}\mathbf{R}_{j+2} \dots {}^{i-1}\mathbf{R}_i$, 来计算 ${}^j\mathbf{R}_i$ 。

步骤 4 计算末端效应器的位置 $\mathbf{p}_n(j)$; 若 $\mathbf{p}_n(j) = \mathbf{p}_d$, 则捕捉成功,转步骤 10。

步骤 5 计算 $\mathbf{v}_d(j) = [\mathbf{p}_d - \mathbf{p}_n(j)]/\Delta t$, $\dot{\theta}_M(j) = (J^*)^+ \mathbf{v}_d(j)$, 若 $|\dot{\theta}_M(j)| > \dot{\theta}_{M \max}$, 则 $\dot{\theta}_M(j) = |\dot{\theta}_{M \max}|$, 式中 $\dot{\theta}_{M \max}$ 为关节的最大运动角速度; 计算 $\dot{\mathbf{v}}_d(j) = [\mathbf{v}(j)_d - \mathbf{v}_d(j-1)]/\Delta t$, $\dot{\mathbf{p}}_n(j) = [\mathbf{p}_n(j) - \mathbf{p}_n(j-1)]/\Delta t$, 根据式(8-38)计算 $\ddot{\theta}_M(j)$, 若 $|\ddot{\theta}_M(j)| > \ddot{\theta}_{M \max}$, 则 $\ddot{\theta}_M(j) = \ddot{\theta}_{M \max}$, 式中 $\ddot{\theta}_{M \max}$ 为关节的最大运动角加速度。

步骤6 用式(8-7)、式(8-9)和式(8-11),计算 $\bar{\omega}_i$ 、 \bar{p}_i 和 $\dot{\bar{r}}_i$;用式(8-30)~式(8-31)计算矩阵 M_1 、 M_2 、 L_o 和 L_w ;根据式(8-29)求解卫星本体的速度 $v_0(j)$ 和角速度 $\omega_0(j)$;用式(8-6)、式(8-8)和式(8-10),计算机械手各连杆的角速度 ${}^i\omega_i(j)$ 、速度 ${}^i\dot{p}_i(j)$ 和其质心速度 ${}^i\dot{r}_i(j)$ 。

步骤7 根据式(8-16)、式(8-18)和式(8-20),计算各连杆中间变量值 ${}^i\bar{\epsilon}_i$ 、 ${}^i\bar{\alpha}_i$ 和 ${}^i\bar{\alpha}_{c_i}$;用式(8-33)计算 F_a 、 F_y ;根据式(8-32)求解卫星本体的加速度 $\alpha_0(j)$ 和角加速度 $\epsilon_0(j)$ 。

步骤8 根据式(8-15)、式(8-17)和式(8-19),计算各连杆的角加速度 ${}^i\epsilon_i(j)$ 、加速度 ${}^i\alpha_i(j)$ 和质心的加速度 ${}^i\alpha_{c_i}(j)$;用式(8-22)~式(8-23),计算作用在各连杆上的力 if_i 和力矩 in_i ;用式(8-24)~式(8-26),计算各关节的驱动力矩 $\tau_i(j)$ 。

步骤9 按照 $\ddot{\theta}_M(j)$ 、 $\dot{\theta}_M(j)$ 和 $\tau_i(j)$ 驱动各关节运动;并计算FFSR的新状态:

(1) 卫星本体和末端效应器的新位置: $p_0(j+1)$ 、 $p_n(j+1)$;

(2) 卫星本体新姿态角 $\theta_0(j+1) = \theta_0(j) + \Delta\theta_0$;

(3) 机械手各关节的角 $\theta_M(j+1) = \theta_M(j) + \dot{\theta}_M(j)\Delta t$;令 $j = j+1$ 并转步骤5。

步骤10 算法结束。

8.3 计算机仿真

1. 关节力矩递推算法仿真

为了验证提出的FFSR力矩递推算法的有效性,不失一般性,针对一种典型的单臂型二自由度FFSR模型进行了计算机仿真研究。本仿真软件在PC机上用Visual C语言开发。FFSR仿真模型如图8-1所示,表8-1给出了模型的物理参数。设给定关节角 θ_1 、 θ_2 的变化规律分别为: $\theta_1 = 45^\circ \cdot \sin(\pi \cdot t/6) + 45^\circ$, $\theta_2 = -45^\circ \cdot \cos(\pi \cdot t/6) - 45^\circ$,式中 $0 \leq t \leq 4$ 。分别将 θ_1 、 θ_2 对时间求一阶、二阶导数可得关节1、2的角速度和角加速度。图8-2是所给定关节角 θ_1 、 θ_2 的变化轨迹。图8-3~图8-5是卫星本体的位姿变化及其所受力矩时间历程。图8-6是关节的驱动力矩。图8-7是FFSR动态仿真过程。计算机仿真结果验证了书中提出力矩递推算法的有效性,该力矩算法的计算量的量级为 $O(n)$,也即与关节自由度成正比。

表8-1 FFSR仿真模型参数表

连杆号 i	a_i/m	B_i/m	l_i/m	m_i/m	$I_i/kg \cdot m^2$
0		0.075	0.075	12.2	0.218
1	0.127	0.073	0.2	0.4	0.0059
2	0.095	0.105	0.2	0.375	0.00525

2. 基于关节驱动力矩捕捉目标仿真

为了验证提出的FFSR关节力矩求解算法及其基础上的捕捉目标控制算法的有效性,针对6自由度机械手的FFSR,在SGI/O2工作站平台上进行了计算机仿真。仿真软件采用IRIS Performer可视化编程语言开发。本软件开发的基本过程为:FFSR几何建模→真实感物理建模→运动学与逆动力学建模→捕捉目标控制→实时仿真。表8-2、表8-3给出了FFSR仿真模型的D-H运动学参数和质量、惯量等动力学参数。FFSR进行目标捕捉

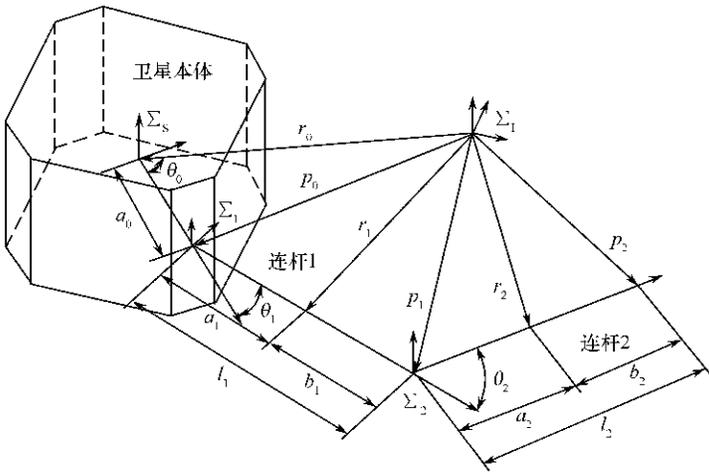


图 8-2 FFSR 仿真模型

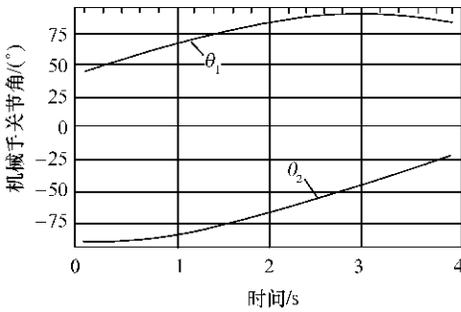


图 8-3 关节角变化时间历程

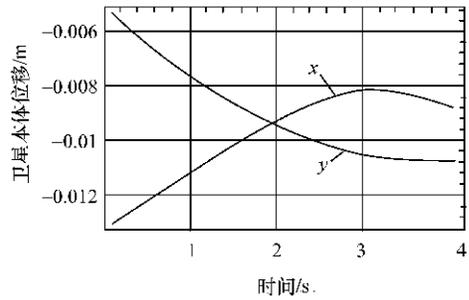


图 8-4 本体位移时间历程

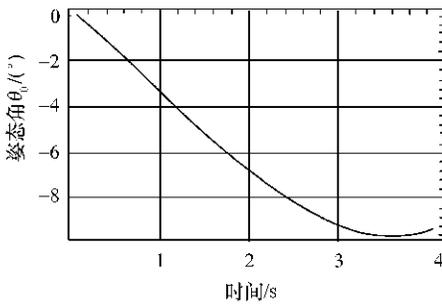


图 8-5 本体姿态角时间历程

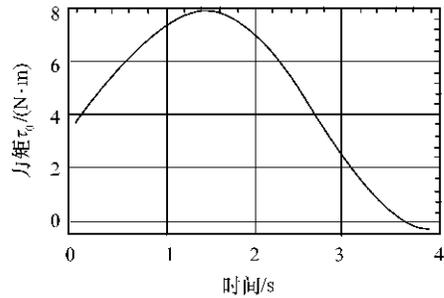


图 8-6 本体所受力矩时间历程

时部分有关仿真结果如下:图 8-8 是 FFSR 捕捉目标的三维图形仿真;图 8-9 是本体的姿态角和位置变化的时间历程;图 8-10 是各关节角的时间历程;图 8-11 是各关节的驱动力矩。计算机仿真结果验证了提出算法 8-2 的有效性。

表 8-2 FFSR 模型的 D-H 运动学参数

关节 i	1	2	3	4	5	6

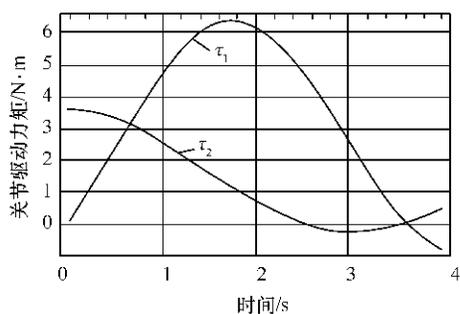


图 8-7 关节驱动力矩时间历程

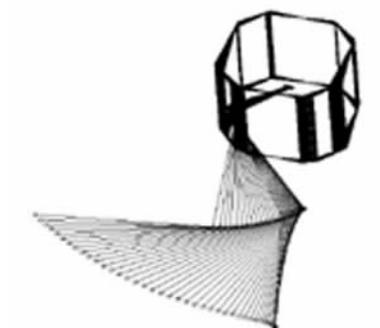


图 8-8 FFSR 动态仿真过程

扭角	α_{i-1}/rad	0	$-\pi/2$	0	$-\pi/2$	$\pi/2$	$-\pi/2$
杆长	l_{i-1}/m	0.3	1.2	2.0	0.8	0.25	0.2
偏距	d_i/m	0.0	0	0.3	-0.3	0.0	0.0
关节角	θ_i/rad	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6

表 8-3 FFSR 动力学参数

链杆号 i		0	1	2	3	4	5	6
m_i/kg		250.0	15.0	12.0	10.0	5.0	2.5	2.0
b_i/m	(x)	0.0	0.0	-0.12	0.0	0.0	0.0	0.0
	(y)	0.0	0.07	0.0	0.0	0.05	0.0	0.0
	(z)	-0.2	0.0	0.0	0.07	0.0	0.05	-0.04
${}^i I_i$ / $\text{kg} \cdot \text{m}^2$	(1,1)	12.5	0.158	0.025	0.095	0.025	0.01	0.007
	(2,1)	4.0	0.0	0.0	0.0	0.0	0.0	0.0
	(3,1)	8.0	0.0	0.0	0.0	0.0	0.0	0.0
	(2,2)	12.5	0.092	0.225	0.095	0.016	0.01	0.007
	(3,2)	5.0	0.0	0.0	0.0	0.0	0.0	0.0
	(3,3)	12.5	0.158	0.225	0.041	0.025	0.004	0.005



图 8-9 FFSR 捕捉目标的三维图形仿真

(a) 初始状态; (b) 终止状态。

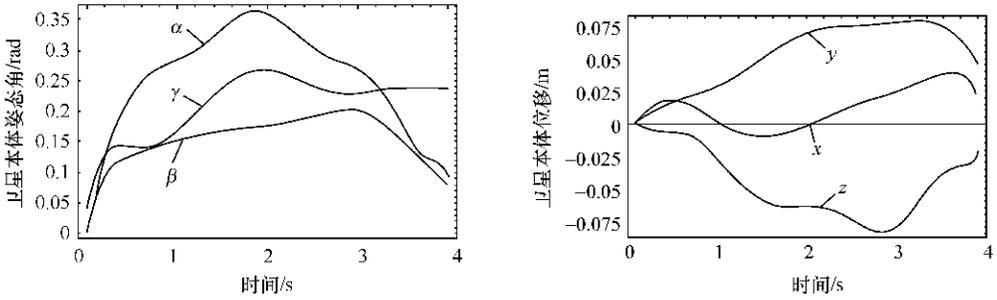


图 8-10 本体的姿态角和位置变化的时间历程

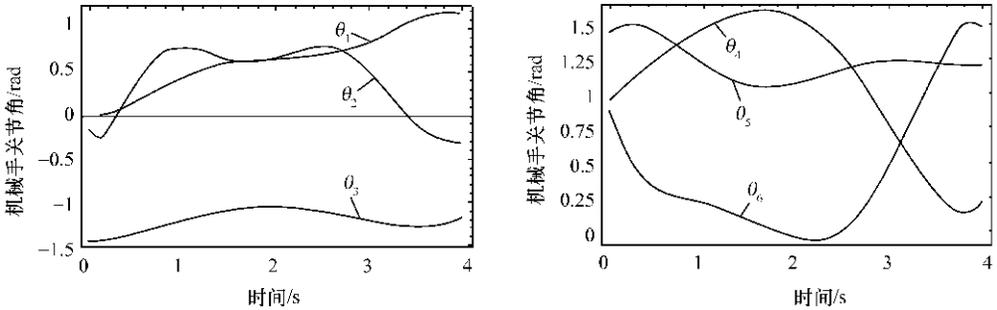


图 8-11 FFSR 机械手关节角的时间历程

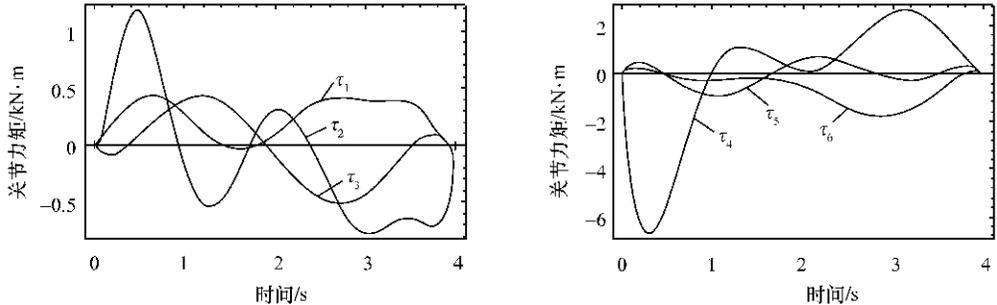


图 8-12 FFSR 机械手关节的驱动力矩时间历程

8.4 小 结

本章讨论了基于关节驱动力矩的 FFSR 运动控制算法。为此,首先进行了 FFSR 的运动学和动力学分析,在此基础上建立了计算 FFSR 机械手各连杆的速度和加速度的递推公式。其次基于 Luh 等人针对地面机器人提出的牛顿-欧拉动力学算法,开发了适用于 FFSR 的关节驱动力矩的求解算法。最后,结合基于广义雅可比矩阵的分解运动加速度控制法(RMAC)提出了一种捕捉目标的力矩控制算法。

第九章 FFSR 模糊运动控制算法

在机器人运动控制中,一个典型的任务是沿着一条给定的(期望)路径移动机器人的末端效应器。末端效应器的位置 $X(t)$ 和关节角 $\Theta(t)$ 之间的关系可以用式(9-1)或式(9-2)表示:

$$X(t) = F[\Theta(t)] \quad (9-1)$$

$$\Theta(t) = F^{-1}[X(t)] \quad (9-2)$$

式(9-1)和式(9-2)分别是机器人的正运动学和逆运动学方程。对于现有的机器人,可以得到和估价正向运动学模型(9-1);但是,逆运动学模型(9-2)代表了 $X(t)$ 和 $\Theta(t)$ 之间的很复杂的映射关系。有些情况下,逆运动学模型(9-2)可能构造不出来,特别是对冗余机器人更难。

已经有人提出几种方法求解冗余和逆运动学问题,其中有最优化约束技术,即通过它可以得到惟一的逆运动学解。但是这种方法也给处理器带来多余的计算量。除了计算量大外,传统的求解逆运动学方程的数学方法又产生了另外一个不期望得到的副产品:运动学奇异点,这是由不正常的矩阵的逆矩阵引起的。

通过查询模糊规则,一些很复杂的、定义不好的系统都可以很好地运行。模糊规则是反映人类对系统的知识和经验的一系列语言陈述。模糊逻辑控制已成功应用于蒸汽机控制、自动化火车操纵系统、升降机控制、核反应堆控制、汽车自动变速控制等领域中。在其他很多应用中,模糊逻辑控制与传统控制方法相比也产生了更好的控制效果。

本章提出了一种基于模糊规则的机器人运动控制方法。采用分级地安排模糊规则以减少控制机器人所需的模糊规则数。

9.1 模糊逻辑和分级模糊控制器

与常规的解析控制器不同,基于模糊规则的控制器的规则构成,这些规则规定只有满足一些特殊的条件时,才能开始某一动作。

在常规的规则集中,规则 i 是下面这种形式:

$$\text{If}(x_1 \text{ is } a_{1j}, x_2 \text{ is } a_{2j}, \dots, x_n \text{ is } a_{nj}) \text{ then (output is } b_j) \quad (9-3)$$

式中, $x_1, x_2, x_3, \dots, x_n$ 是一些系统变量, $a_{ij} \in A(i = 1, 2, \dots, n; j = 1, 2, \dots, k)$; $b_j \in B(j = 1, 2, \dots, k)$ 。

A 和 B 是由模糊集中的元素组成的集合,它们可能相同。例如: $A = B = (PB, PM, PS, PZ, Z, NZ, NS, NM, NB)$ 是模糊集合,它们分别表示正大、正中、正小、正零、零、负零、负小、负中和负大。模糊集是没有明显边界的集合。换句话说,模糊集合与含糊性是相关的,它是由隶属函数决定的。

模糊集合 a_{ij} 和 b_j 的隶属函数可以用很多种方式定义:如三角形的、指数型的等等。隶属函数的选择取决于实际情况。把模糊集 a_{ij} 的隶属函数记为 $u_{a_{ij}}(x)$ 。然后,用式(9-3)中的模糊规则,模糊控制器的输出可以计算得到:

$$u_j = u_{a_j}(X) * y_j \quad (9-4)$$

式中, y_j 满足

$$u_{b_j}(y_j) = 1 \quad (9-5)$$

并且,对于 $i = 1, 2, \dots, n$, 有

$$u_{a_j}(X) = \min(u_{a_j}(x_i)) \quad (9-6)$$

由整个规则集产生的输出为:

$$u_0 = \frac{\sum_{i=1}^n u_{a_i} * y_i}{\sum_{i=1}^n u_{a_i}} \quad (9-7)$$

在式(9-7)中,所有的规则都用来计算输出 u_0 ,这使得计算复杂而且费时间。

定理 9-1 如果相同变量数为 n ,模糊集合数为 m ,那么一个完整的规则集有 m^n 条不同的规则。

由上可见,规则总数是系统变量数的指数函数。对应多维系统(n 很大),可能很难构造一个基于规则的模糊控制器。为了克服这个问题,提出了一个分级模糊控制结构,使得在第一级中,选择最起作用的参数作为系统变量,其次要作用的参数在第二级中作为系统变量等等。

第一级中的规则形式如下:

$$\text{If } x_1 \text{ is } a_{1,1}, x_2 \text{ is } a_{1,2}, \dots, x_n \text{ is } a_{1,n} \text{ then output } y_1 \text{ is } b_1 \quad (9-8)$$

第 i 级中的规则形式如下:

$$\text{If } x_{N_i+1} \text{ is } a_{N_i,1}, \dots, x_{N_i+n_i} \text{ is } a_{N_i,n_i} \text{ and } y_{i-1} \text{ is } b_{i-1} \text{ then output } y_i \text{ is } b_i \quad (9-9)$$

式中, $N_i = \sum_{j=1}^{i-1} n_j \leq n$, n_j 是在第 i 级中所用到的系统变量。这里 x_j ($j = 1, 2, \dots, n$) 是系统变量, y_i ($i = 1, 2, \dots, l-1$) 是第 i 级的输出,并且在第 $i+1$ 级作为变量。

在这种层次中,第一级给出了一个近似输出 y_1 ,它由第二级规则集进行修正。第二级的变量包括第一级的近似输出 y_1 和式(9-9)中的系统变量。这一过程在后续层次中重复执行。

定理 9-2 对于由 n 个系统组成的层次结构,如果 t 是每个层次中包含的变量数目(其中包括其上一层的输出变量),那么规则总数 k 为:

$$k = ((n-t)/(t-1) + 1) * m^t \quad (9-10)$$

一个完整的规则集中规则数,可以通过层次结构,由常规情况下的 n 的指数函数减少到 n 的线性函数。这使得计算机运行整个规则集的计算量大大地减小了。

9.2 基于模糊规则的机器人运动控制算法

与常规数学方法不同,模糊控制器输出信号通过查找模糊规则来获得。这些模糊规则

是人类知识和经验的反映。这项研究的重点是如何构造正确的模糊规则。人可以不用许多精确的数学知识就能够灵巧地控制自己的手臂，而机器人控制需要复杂的数学公式。这说明非数学化的模糊逻辑也应该可以用于机器人运动控制。

为了方便模糊控制器设计问题的研究，首先考虑一个二自由度平板机器人，然后，把这个方案推广到运动学冗余的三自由度平板机械手。

9.2.1 二自由度机器人的基于模糊规则的控制设计

假想机器人从初始位置 (x_0, y_0) 运动到期望位置 (x_d, y_d) ，如图 9-1 所示，其中 a_{10} 和 a_{20} (a_{1d} 和 a_{2d}) 分别为关节 1 和关节 2 的初始(期望)角度。初始时，机器人臂长为 r_0 ，期望长度为 r_d ，它们是末端效应器与原点的相应距离。将连杆 1 和连杆 2 的长度分别记为 l_1 和 l_2 ，当前的机器人臂长由式(9-11)给出。

$$r = \sqrt{x^2(t) + y^2(t)} \quad (9-11)$$

$$r^2 = l_1^2 + l_2^2 + 2l_1 l_2 \cos\alpha_2 \quad (9-12)$$

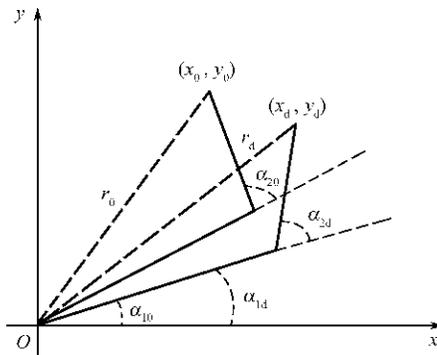


图 9-1 二自由度机器人的初始和期望位姿

因为 r 只是 α_2 的函数，可以只改变 α_2 就能得到期望臂长 r_d ，得到 r_d 的变化与所需的 α_2 变化之间的数学关系如下：

$$\Delta^2 r = r_d^2 - r_0^2 \quad (9-13)$$

$$\Delta^2 r = l_1^2 + l_2^2 + 2l_1 l_2 \cos\alpha_{2d} - (l_1^2 + l_2^2 + 2l_1 l_2 \cos\alpha_{20}) = \quad (9-14)$$

$$2l_1 l_2 (\cos\alpha_{2d} - \cos\alpha_{20}) = \quad (9-15)$$

$$2l_1 l_2 [\cos(\alpha_{20} + \Delta\alpha_2) - \cos\alpha_{20}] = \quad (9-16)$$

如果 $\Delta\alpha_2$ 很小，那么式(9-16)就变为：

$$\begin{aligned} \Delta^2 r &= 2l_1 l_2 (\cos\alpha_{20} - \sin\alpha_{20} * \Delta\alpha_2 - \cos\alpha_{20}) = \\ &= -2l_1 l_2 \sin\alpha_{20} * \Delta\alpha_2 \end{aligned} \quad (9-17)$$

在这个算法中，用 Δr^2 而不用 Δr 作为模糊规则的输入，使用 Δr^2 可以减小计算量，因为 r 通常是通过 x^2 加 y^2 计算出的。在通过变化 α_2 获得期望的 r_d 后，变化 α_1 以旋转机器人

手臂到达期望位置。这个过程的图像描述由图 9-2 给出。其中 $\theta = \angle EOx$; E 、 O 和 x 是末端效应器, 原点和 x 轴的正方向上的点, 而且 $\Delta\theta = \theta_d - \theta_0$, θ_d 和 θ_0 分别代表期望关节角和初始关节角。

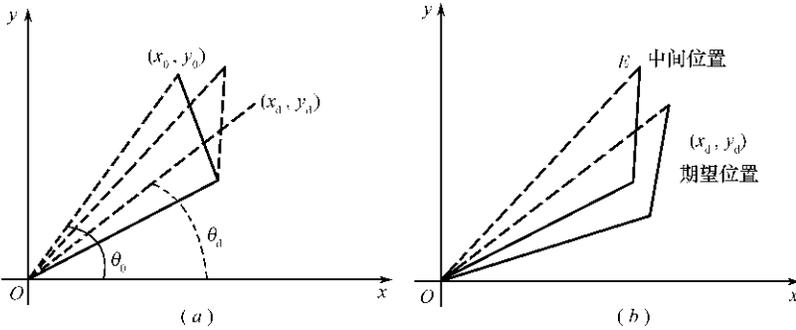


图 9-2 过程描述

(a) 通过改变 a_2 使机器人手臂伸展到期望长度; (b) 通过改变 a_1 使机器人手臂旋转到期望位置。

基于这一思想, 可以把模糊规则分成两级以构造模糊控制器, 如图 9-3 所示。通过使用作者提出的这种两级层次结构和分级模糊控制技术, 开发模糊控制器所需的规则总数大大减少了。

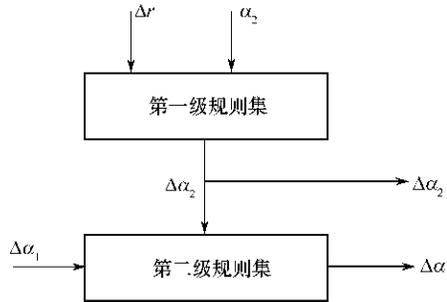


图 9-3 两级分层规则

一般的规则可能是如下形式:

$$\begin{aligned} &\text{If}(\alpha_{10} \text{ is } x_1) \text{ and } (\alpha_{20} \text{ is } x_2) \text{ and } (x_d \text{ is } x_3) \text{ and } (y_d \text{ is } x_4) \\ &\text{then } (\alpha_{10} \text{ 改变 } \Delta\alpha_1) \text{ and } (\alpha_{20} \text{ 改变 } \Delta\alpha_2) \end{aligned} \quad (9-18)$$

在这个模糊规则例子中, $x_1, x_2, x_3, x_4, \Delta\alpha_1$ 和 $\Delta\alpha_2 \in A$ 是一些语言变量值。在第一级的模糊规则形式是:

$$\begin{aligned} &\text{If}(\text{机器人的臂长需要改变 } x_1) \text{ and } (\text{当前的关节角 } \alpha_2 \text{ 为 } x_2) \\ &\text{then}(\alpha_2 \text{ 改变 } \Delta\alpha_2) \end{aligned} \quad (9-19)$$

第二级中的模糊规则是:

$$\begin{aligned} &\text{If}(\alpha_2 \text{ 的变化为 } \Delta\alpha_2) \text{ and } (\text{期望的角度变化 } \Delta\theta \text{ 为 } x_2) \\ &\text{then}(\alpha_1 \text{ 改变 } \Delta\alpha_1) \end{aligned} \quad (9-20)$$

第一级和第二级中规则构成了机器人运动控制的基础。在模糊控制器中所开发的模糊规则输出了一个增量关节命令到各个机器人关节, 还输出了从实际末端点到期望末端点的偏差。实际的笛卡儿末端点是通过在关节角上运用正向运动学方程获得的。期望路径

是用户定义的。

9.2.2 三连杆 FFSR 的基于模糊规则的控制设计

图 9-4 中给出了一个有运动学冗余的三连杆空间机器人的位姿,这里假设空间机器人的本体为一个质量集中在一点的一个连杆,那么三连杆空间机器人就转化为一个四连杆的无固定底座的机器人系统。图中的 I_xOI_y 为惯性坐标系, xOy 为原点在各关节 1 的坐标系,这个坐标系会随着机器人各关节的转动而做相应的旋转;另外,图中的第一个连杆(即机器人的本体)是不能主动转动的,它在其余的连杆转动时才会转动; α_i 是关节 i ($i = 1, 2, 3$) 相对于前一个连杆的关节角, α_0 是本体的姿态角,可以通过传感器或摄像机获得这些角度值, r 是末端效应器与机器人的关节 1(与机器人本体相连接的关节)间的距离; x_e 和 y_e 为末端效应器的笛卡儿坐标。

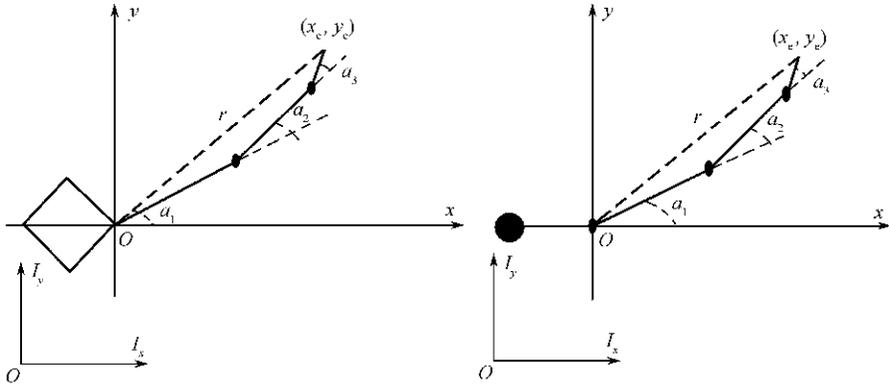


图 9-4 三连杆 FFSR 的描述

在这个实验中,在每个采样点通过查询模糊规则集来改变关节角控制机器人沿给定路径运动。基于 9.2.1 中的讨论,可知机器人的典型运动由以下两部分组成。

- 动作 1 伸展或折叠机器人手臂;
- 动作 2 旋转机器人手臂。

动作 1 涉及到关节角 α_2, α_3 的变化,而动作 2 涉及到关节角 α_1 的变化。进一步研究表明,如果只考虑末端效应器位置,可以通过改变 α_2, α_3 中的一个值来完成动作 1。因此,此时相当于机器人只有两个关节,关节 1 完成动作 1,关节 2 完成动作 2,相应的伪连杆的长度 L_1 和 L_2 分别由式(9-21)和式(9-22)给出。

$$L_1 = \sqrt{x_i^2 + y_i^2} \quad (9-21)$$

$$L_2 = \sqrt{(x_i - x)^2 + (y_i - y)^2} \quad (9-22)$$

因此,可以像对一个二自由度机器人一样控制一个三自由度机器人。于是模糊规则集也与 9.2.1 中的形式类似。

通过查找模糊规则值,在每个采样点只改变三个关节角中的两个。可以在另外的采样时刻改变其他的关节角。采用下面的策略来确定冗余机器人中需要改变的关节角。如果伸展机器人手臂,那么应旋转角度绝对值最大的关节,反之亦然。这样,可以避免每个关节碰到它的奇异点,从而使得伸展(折叠)机器人手臂时角度变化最小。

前面已经提到,运动学解的不惟一性阻碍了运动学冗余机器人的实时运动控制。由于这种基于模糊规则的方法不需要逆运动学模型,所以运动学冗余机器人有没有惟一解就不重要了。

9.2.3 基于模糊逻辑的 FFSR 路径规划算法

基于模糊逻辑的 FFSR 的路径规划算法如下。

步骤 1 确定机器人的初始状态信息:设期望目标点在惯性坐标系中为 P_{ex} , 根据机器人的几何参数和坐标变换原理计算期望点在机器人的关节 1 的坐标系中为 $P_{ex}^1(0)$, 机器人本体位置为 $P_0(0)$, 本体姿态为 $\phi_0(0)$, $\phi_M(0)$ 为关节角向量。

步骤 2 令 $i = 0$ 。

步骤 3 计算机器人末端效应器在关节坐标系 1 中的位置 $P_e^1(i)$ 。

步骤 4 若 $|P_e^1(i) - P_{ex}^1(i)| \leq err$, 则捕捉成功, 转步骤 11, 式中 err 为误差允许范围。

步骤 5 计算目标点到关节坐标系 1 的原点的距离 $r_d(i)$, 以及与这个坐标系的 x 轴的夹角 $\theta_d(i)$; 计算机器人末端效应器到关节坐标系 1 的原点的距离 $r(i)$, 以及与这个坐标系的 x 轴的夹角 $\theta(i)$ 。

步骤 6 计算 $\Delta\theta = \theta_d(i) - \theta(i)$, $\Delta r = r_d(i) - r(i)$; 然后对它们进行模糊量化, 作为分层模糊控制器的输入变量, 经过模糊推理和反模糊化后得出输出变量 $\Delta\alpha_1, \Delta\alpha_2$ 。

步骤 7 按照 $\alpha_1(i), \alpha_2(i), \alpha_3(i)$ 驱动机器人各关节转动。

步骤 8 计算机器人的新状态:

(1) 计算机器人本体的新位置 $P_0(i+1) = P_0(i) + \Delta P_0$ 。

(2) 计算机器人本体姿态角 $\phi_0(i+1) = \phi_0(i) + \Delta\phi_0$ 。

(3) 计算机器人本体的姿态角及各关节的新的关节角 $\alpha_j(i+1) = \alpha_j(i) + \Delta\alpha_j$, 式中 $j = 0, \dots, 3$ 。

步骤 9 计算机器人末端效应器在新的关节坐标系 1 中的位置 $P_e^1(i+1)$ 。

步骤 10 $i = i + 1$, 转步骤 4。

步骤 11 算法结束。

在实际运用中,包括本体的姿态角在内的各种角度的变化可以通过传感器获得。在计算机仿真中,本体的姿态角的规划可以通过广义雅可比矩阵 GJM 的推导过程获得。在推导 GJM 时,有: $I_S \dot{\phi}_0 + I_M \dot{\phi}_M = 0$ 成立, 式中 $\dot{\phi}_0$ 为本体的角速度, $\dot{\phi}_M$ 为关节的角速度, 在离散的情况下, 此式就可写为:

$$I_S \Delta\phi_0 + I_M \Delta\phi_M = 0 \quad (9-23)$$

在这个算法中,由于机器人的手臂的运动对本体产生了姿态干扰,所以建立在关节 1 上的坐标系的原点在惯性坐标系中是变化的,期望目标点在惯性坐标系中是不变的,但在这个坐标系中是变化的。根据式(9-23) 求出每个时刻本体的姿态角,实时更新期望目标点在关节 1 坐标系中的表示,实现空间机器人的规划目标。

在实际应用中,这个算法与以前的一些算法(如基于 GJM 的分解运动速度控制算法等)相比,将会大大地减少计算量。这是因为,以前的算法中,都要求解 GJM 的广义逆矩阵来计算各关节的角速度,这需要很大的计算量。而在这个算法中,各个关节的转角是根

据模糊推理计算出来的,不需要计算 GJM 的逆矩阵,有利于实现机器人系统的实时控制。

9.3 计算机仿真

基于模糊逻辑的 FFSR 的路径规划算法在 PC 机上用 Matlab 进行了仿真。机器人的本体为边长为 $l/2$ 的正方形;质量为 m_0 ;机器人的连杆长度分别为 $l, l, l/2$;各连杆的质量分别为 m_1, m_2, m_3 ,其中 $m_1 = m_2 = 2m_3 = m_0/10, l = 0.4\text{m}, m_1 = 0.5\text{kg}$ 。图 9-5 给出了机器人手臂运动到给定点的问题的描述,图中 $I_x O I_y$ 为惯性坐标系,机器人末端的初始坐标为 (x_e, y_e) ,期望的末端坐标为 (x_d, y_d) 。

图 9-5 表示了当指定了空间机器人工作空间中的一个目标点 $P_{ex}(x_d, y_d)$ 后,应用上面所提出的算法,FFSR 到达了给定的目标点。图 9-6 表示 FFSR 在接近目标点的过程中,机器人本体的中心位置的变化。实验表明 FFSR 末端效应器的最终位置与所指定的目标点之间的距离在给定误差范围之内。

经过实验表明,只要目标点在空间机器人的工作空间内,无论初始误差有多大,稳态误差都在给定误差限之内,这证明所提出的算法具有很好的鲁棒性。

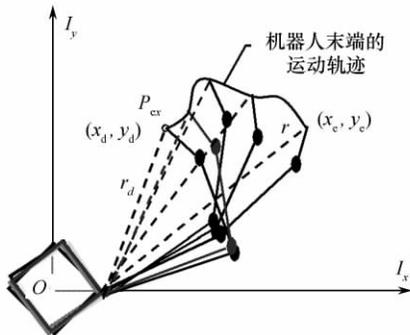


图 9-5 机器人末端运动到给定点

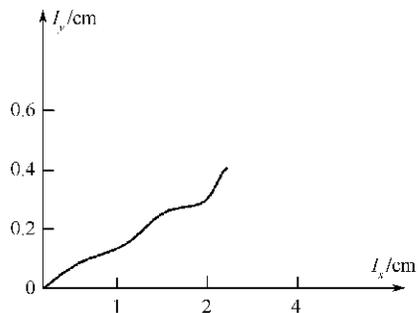


图 9-6 本体中心变化曲线

9.4 小 结

本章提出了一种基于模糊规则的机器人运动控制方法来减少与常规数学算法相关的逆运动学问题的计算复杂性。用层次化方法来构造模糊规则以减少完整的模糊规则集所需的规则数,并基于这种分级模糊规则提出了一种机器人路径规划算法,对所提出的算法进行分析,可以看出,所提出的算法由于不需要计算 GJM 的广义逆矩阵,所以大大地减小了计算量,有利于实现机器人的实时路径规划;而且还避免了奇异点,提高了机器人的控制效率。计算机仿真还表明,运用所提出的算法,FFSR 可以到达指定的目标点,实验还证明了这种基于模糊规则的控制器具具有很强的鲁棒性。

第十章 FFSR飞行轨迹优化

在空间作业中,FFSR 的运动可以分为两部分,即机器人本体的飞行运动和机械手的操作运动。FFSR 携带的推进器能够使之以期望的速度和姿态达到期望的位置,它的工作空间是不受限制的。因此,FFSR 能够飞行到很远的工作地点,执行维修、装配或回收失效卫星等作业。在 FFSR 回收失效卫星时,若目标与 FFSR 不在同一个轨道上,FFSR 需要变轨,然后在同一轨道上接近和捕捉目标。本章主要研究在同一轨道上接近目标的情况。为了扩展 FFSR 的工作空间,提高 FFSR 的工作效率并延长在轨寿命,对其飞行运动轨迹的研究具有十分重要的意义。

本章主要讨论 FFSR 飞行轨迹优化算法,同时考虑时间和燃料两种因素,以寻找在一定条件下 FFSR 的最佳飞行轨迹。首先介绍如何建立优化模型,其次讨论采用的优化方法,最后通过计算机仿真验证所提出的优化算法的正确性。

10.1 FFSR 飞行轨迹优化模型

对于 FFSR 飞行轨迹的研究较为典型的是等加速运动、匀速运动和等减速运动组成轨迹,它是在规定时间内飞行一定距离消耗燃料最少的运动轨迹。然而,时间效率和燃料消耗对于 FFSR 来说都是至关重要的,因此 FFSR 飞行轨迹优化应该同时考虑时间和燃料两种因素。

FFSR 飞行轨迹的优化就是寻找一条机器人本体飞行运动轨迹,使机器人以期望的速度和姿态从一点运动到另一点。该轨迹应该包含特定的几何因素和时间因素。几何因素必须保障机器人避开所有的障碍物,时间因素则描述机器人沿轨迹运动过程中每一时刻的速度和加速度,应该保证在运动过程中的速度和加速度不超过机器人的驱动能力。本节以飞行时间短和燃料消耗少为优化目标建立多目标优化模型,建立了目标函数,并选择优化参数和约束条件。主要建立接近静止目标轨迹优化、截击运动目标轨迹优化和多个 FFSR 飞行轨迹优化模型。

10.1.1 单 FFSR 接近静止目标飞行轨迹优化

为接近静止目标,FFSR 以一定的初始速度从出发点飞至会合点。到会合点处机器人处于静止状态,即机器人的速度和加速度均为零,此时应保障目标在机器人的工作空间范围内。FFSR 接近静止目标飞行轨迹如图 10-1 所示。

1. 优化参数

机器人的飞行轨迹是连续的,可以把它离散化,即分成若干段,每一段可以用一系列参数来描述,如距离、速度、加速度、驱动力和运动时间等。机器人将沿着每一段轨迹从初

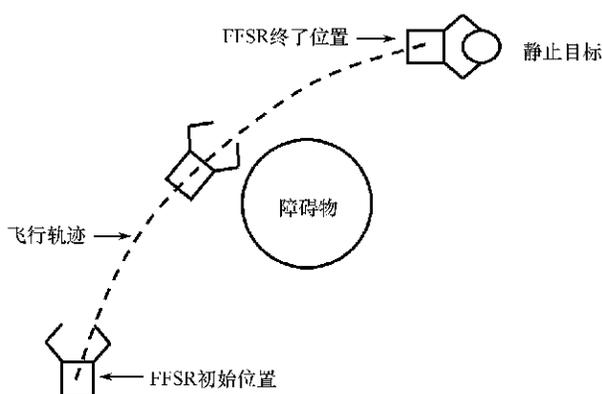


图 10-1 FFSR 接近静止目标飞行轨迹

始点向前运动,避开所有障碍物,并在目标点以期望的终了速度结束运动。轨迹离散化概念图如图 10-2 所示。

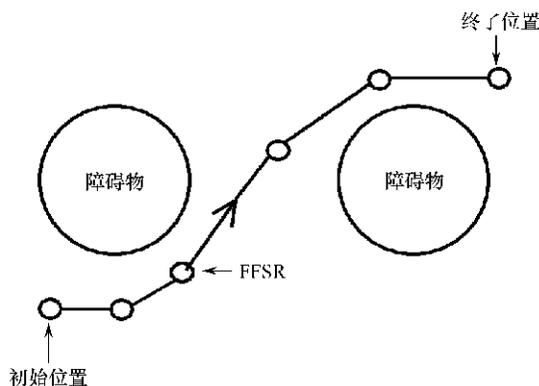


图 10-2 轨迹离散化概念图

燃料用量与机器人所受的驱动力之间存在着线性关系,而燃料用量又是优化的目标,所以可以选择驱动力作为优化参数,从而可以简化优化算法、提高收敛速度。由于运动时间也是优化的目标,所以可以选择每段运动时间作为优化参数。本章讨论的优化算法选择每一段轨迹上的驱动力和运动时间作为优化参数。

此处定义两个坐标系:惯性坐标系 (x, y, z) 和机器人本体坐标系 (x', y', z') 。

如果把整个飞行轨迹分成 n 段,机器人在三维空间内运动,则有如下固定参数,即初始点的位置 (x_0, y_0, z_0) 和速度 (v_{0x}, v_{0y}, v_{0z}) ,终点的位置 (x_f, y_f, z_f) 和速度 (v_{fx}, v_{fy}, v_{fz}) 。优化参数分别选择每段上机器人在各个方向上所受的驱动力 $(F_{1x'}, F_{1y'}, F_{1z'}, \dots, F_{nx'}, F_{ny'}, F_{nz'})$ 和每段的运动时间 $(\Delta t_1, \Delta t_2, \dots, \Delta t_n)$ 。本章中假设每段内的驱动力是恒定的,每段的运动时间相等,即 $\Delta t_1 = \Delta t_2 = \dots = \Delta t_n$ 。

机器人在三维空间内有一个旋转运动。如果把产生旋转运动的驱动力也作为优化参数,则可以优化出一系列的旋转运动,从而得到性能更好的优化轨迹。但同时也会使问题复杂化,如旋转驱动力与移动驱动力相互干扰使算法复杂化、收敛速度变慢。因此本章不选择旋转驱动力作为优化参数,而只用一个参数代表优化的机器人的最终方向与初始方

向的夹角 $\Delta\theta$ 。机器人通过轨迹上的每个分割点时,分别旋转角度 $\Delta\theta_i$ ($\Delta\theta = \Delta\theta_i \times (n-1)$)。这样,机器人只需很小的旋转加速度和旋转速度就可以实现旋转运动,最终机器人将朝向优化的方向。

在第 $i+1$ 点的位置可由第 i 点的位置和速度及第 i 段的加速度和运动时间得到,即:

$$s_{i+1} = s_i + v_i \Delta t_i + \frac{1}{2} a_i \Delta t_i^2 \quad (10-1)$$

式中, s 是位置向量; v 是速度向量; a 是加速度向量。

在第 $i+1$ 点的速度可由第 i 点的速度及第 i 段的加速度和运动时间得到,即:

$$v_{i+1} = v_i + a_i \Delta t_i \quad (10-2)$$

应用式(10-1)和式(10-2),给定一个参数向量和初始条件,可以计算出运动轨迹的所有信息,通过这些信息可以确定飞行轨迹优化的目标函数和约束方程。

2. 目标函数

对于 FFSR 来说,燃料是至关重要的,因为它是不可再生的,从而决定了机器人的在轨寿命。本章的优化算法把飞行过程中的燃料消耗最少作为优化目标。如果把机器人看成一个质点,在轨迹的每一个阶段在机器人的每个轴线上的燃料用量与所受驱动力的绝对值和运动时间的乘积成正比,那么,整个飞行过程中的燃料消耗可以近似地表示为各阶段机器人所受驱动力的绝对值和运动时间的乘积之和,即:

$$\text{fuel} = \sum_{i=1}^n (|F_{ix'}| + |F_{iy'}| + |F_{iz'}|) \times \Delta t_i \quad (10-3)$$

运动时间对于 FFSR 来说也是一个重要的因素,特别是在时间效率要求较高的工作环境下。所以本章也把飞行给定距离所用的时间最短作为优化目标。

$$\text{time} = \sum_{i=1}^n \Delta t_i \quad (10-4)$$

由于选择燃料用量和飞行时间两个优化目标,所以该优化问题就变成了多目标优化问题。最终的目标函数是燃料用量目标和运动时间目标的线性组合,可用下式表示:

$$f = c \times \text{time} + \text{fuel} \quad (10-5)$$

式中, time 是各阶段所需飞行运动时间之和,用 s 来度量; fuel 是各阶段所受的驱动力与运动时间乘积之和,用 $N \cdot s$ 来度量, c 为时间权重因子。

3. 约束条件

该算法中有如下三种类型的约束条件。

(1) 限制驱动力。机器人所受的驱动力应小于或等于该方向上所能获得的驱动力的最大值。这一约束作用于轨迹的每一段,如果在每一阶段上有三个力 $F_{x'}$ 、 $F_{y'}$ 和 $F_{z'}$,而且轨迹有 n 段,则有 $3n$ 个约束方程。约束方程可以编入一个单列矩阵,所以有如下公式:

$$g_i = \begin{bmatrix} |F_{ix'}| - F_{ix'\max} \\ |F_{iy'}| - F_{iy'\max} \\ |F_{iz'}| - F_{iz'\max} \end{bmatrix} \quad (10-6)$$

式中, $F_{ix'}$ 、 $F_{iy'}$ 和 $F_{iz'}$ 分别为第 i 段轨迹上机器人各个方向上所受的驱动力; $F_{ix'\max}$ 、 $F_{iy'\max}$ 和 $F_{iz'\max}$ 分别为第 i 段轨迹上机器人各个方向上所能获得的最大驱动力。

(2) 机器人避障。约束方程应保证机器人避开所有的障碍物。最典型的的就是保证在轨

迹的分割点处,机器人与障碍物之间的距离大于零。本章中假设障碍物和机器人均为近似圆形,可以减少碰撞检测的次数。要保证分割点到障碍物圆心的距离大于障碍物半径与机器人半径之和,因此避障约束方程可以表示为第 i 点到第 j 个障碍物的距离:

$$g_2 = k_c \cdot (r_c + r_r) - \sqrt{(x_i - x_{c_j})^2 + (y_i - y_{c_j})^2 + (z_i - z_{c_j})^2} \quad (10-7)$$

式中, (x_i, y_i, z_i) 是第 i 点的坐标; $(x_{c_j}, y_{c_j}, z_{c_j})$ 是第 j 个障碍物中心的坐标; r_c 是障碍物的半径; r_r 是机器人的半径; k_c 是避障安全因数, $k_c \geq 1$ 。

(3) 会合静止目标。机器人在运动终点的位置和速度应与轨迹的终了条件相对应。对于会合静止目标的飞行轨迹将限制终点的位置为会合点的位置,而速度为零。因此有 6 个约束方程,它们是:

$$g_3 = \begin{bmatrix} x_f - x_g \\ v_{xf} - 0 \\ y_f - y_g \\ v_{yf} - 0 \\ z_f - z_g \\ v_{zf} - 0 \end{bmatrix} \quad (10-8)$$

式中, (x_g, y_g, z_g) 是会合点的位置。

10.1.2 单 FFSR 截击运动目标飞行轨迹优化

截击并捕捉运动目标是 FFSR 进行空间作业的一项重要内容。首先应该选择会合点,在会合点处应保障目标处于机器人的工作空间范围内。机器人从初始位置以期望的速度和姿态飞至会合点,在会合点处与运动的目标会合,捕捉目标。FFSR 截击运动目标飞行轨迹如图 10-3 所示。因为目标是运动的,所以机器人和目标在会合点处,它们的速度之间应该满足一定的关系。

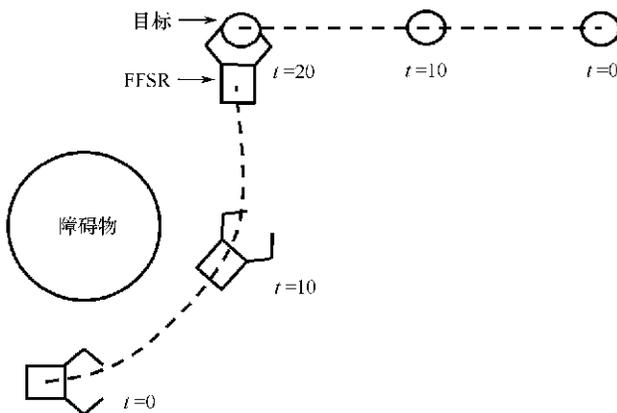


图 10-3 FFSR 截击运动目标飞行轨迹

1. 轨迹生成条件

为了能够准确地会合和捕捉运动目标,必须给出合理的截击轨迹。截击轨迹应保证在会合点处,运动目标处于机械手的工作空间范围内。

对于小质量的目标,本体的截击轨迹可以是指向会合点的简单的一条曲线。捕捉的目标对机器人运动的影响将会很小。而对于大质量的目标,机器人本体在捕捉目标之前必须跟踪运动的物体以减小目标运动对机器人运动产生的影响。因此对于轨迹的生成条件可以总结如下。

- (1) 机械手末端的位置:必须与目标的位置一致。
- (2) 机械手末端的速度:必须与目标的速度一致。
- (3) 机器人本体的位置:必须保证在截击时刻,目标在机器人工作空间之内。
- (4) 机器人本体的速度:当目标质量相对于机器人很大时,必须与目标的速度一致。

通过上面的轨迹生成条件的基本思想,可以得到以下的决定截击轨迹的准则。

准则 10-1 在截击时刻,机械手末端的位置取决于目标的位置和方向。

准则 10-2 如果被捕捉目标的质量很小,不必在截击时刻对本体的运动速度进行任何限制,应限制其位置以保障目标处于机器人的工作空间范围内。

准则 10-3 如果被捕捉目标的质量很大,必须保证目标处于机器人的工作空间范围内,同时机器人本体的运动速度应与目标的运动速度一致。

2. 优化目标函数

由于小质量运动目标的截击轨迹与接近静止目标的飞行轨迹类似,此处不再介绍。在此只介绍截击大质量目标的情况。此时优化数学模型的目标函数和优化参数均与接近静止目标的飞行轨迹一致。

3. 优化约束方程

约束条件首先应限制驱动力和实现机器人避障,因此也选择接近静止目标飞行轨迹优化中约束条件 g_1 和 g_2 。

要截击运动目标,首先要预见目标在截击时刻的位置和速度,因此必须知道目标的运动规律和当前状态。在 FFSR 空间实验中采用 GPS 系统来确定目标的运动规律,一般的地面实验系统用全局视觉系统来确定目标的运动规律,从而实现实时反馈。本章假设目标的运动规律已知,且目标是匀速运动的。在截击时刻,其位置可以表示为:

$$s_{i_0} = s_{c_0} + v_{c_0}(t_{i_0} - t_{c_0}) \quad (10-9)$$

式中, t_{i_0} 是截击时刻; t_{c_0} 是当前时刻; s_{c_0} 是当前位置; s_{i_0} 截击时刻的位置。

在截击时刻,其速度可以表示为:

$$v_{i_0} = v_{c_0} \quad (10-10)$$

式中, v_{c_0} 是当前运动速度; v_{i_0} 是截击时刻的运动速度。

截击运动目标应该考虑机器人相对于目标的状态。截击时刻目标的位置应处于机械手操作的工作空间的中心,即有如下约束方程:

$$g_3 = |s_{i_w} - s_{i_0}| \quad (10-11)$$

式中, s_{i_w} 是截击时刻机器人工作空间中心的位置; s_{i_0} 是截击时刻运动目标的期望位置。 s_{i_w} 可由机器人本体中心位置和机器人的方向来计算。

$$s_{i_w} = s_i + R_w \times A \quad (10-12)$$

式中, s_i 是本体中心的位置; R_w 是本体中心到工作空间中心的距离; A 机器人本体坐标系在惯性坐标系中的旋转矩阵。对于机器人和目标的相对状态,应保证 R_w 的方向与目标的运动方向垂直或平行。

机器人要有效地捕捉目标,截击时刻机器人本体的运动速度应与目标的运动速度一致,因此有约束方程:

$$g_4 = |v_i - v_{io}| \quad (10-13)$$

10.1.3 多 FFSR 飞行轨迹优化

多个机器人协调与合作在 FFSR 空间作业中将发挥越来越重要的作用。经常是多个机器人同时飞至会合点操作同一个较大的目标,此时对其飞行轨迹的研究具有重要的意义。因为它不是单个机器人轨迹优化的简单组合,而是整体的最优化,即以所有机器人到达会合点所用的时间短和燃料消耗少为优化目标。本节研究多个机器人同时接近静止目标的飞行轨迹优化算法。多 FFSR 飞行轨迹如图 10-4 所示。下面介绍 r 个机器人飞行轨迹优化模型。

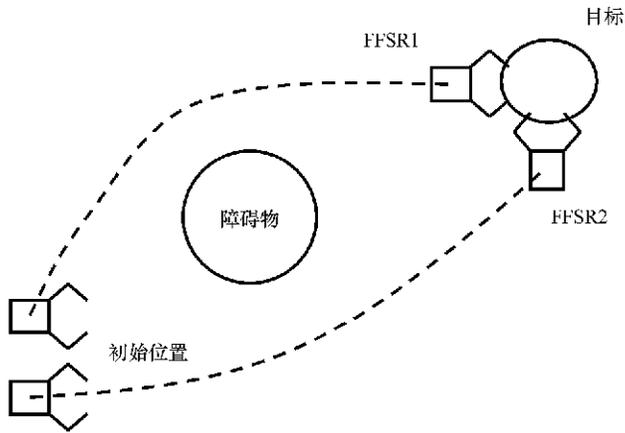


图 10-4 多 FFSR 接近静止目标飞行轨迹

1. 目标函数

对于 r 个 FFSR 系统的飞行轨迹优化,可以把 r 个机器人作为一个整体进行研究,以实现整个系统的最优。参考单个机器人飞行轨迹的优化,选择驱动力和运动时间为优化参数,选择燃料消耗少和运动时间短为优化目标。可以把每个机器人的轨迹都按相同的方式进行离散化,即具有相同的段数和每段相同的持续时间。整个系统燃料消耗为各个机器人燃料消耗之和:

$$\text{fuel} = \sum_{j=1}^r \text{fuel}(j) \quad (10-14)$$

整个系统的运动时间为各个机器人运动时间之和:

$$\text{time} = \sum_{j=1}^r \text{time}(j) \quad (10-15)$$

最终的目标函数是燃料用量目标和运动时间目标的线性组合,其形式如式(10-4)。

2. 约束条件

多个 FFSR 系统飞行轨迹优化,本章只研究接近静止目标的情况,对于截击运动目标可以结合单个机器人截击运动目标的情况简单推广。共有以下四个方面的约束条件。

(1) 限制驱动力。每个机器人所受的驱动力应小于或等于该方向上所能获得的驱动

力的最大值。共有 $3r_n$ 个约束方程：

$$\mathbf{G}_1 = \begin{bmatrix} g_{11} \\ \vdots \\ g_{r1} \end{bmatrix} \quad (10-16)$$

式中, g_{r1} 为式(10-6)中的 g_1 。

(2) 机器人避障。约束方程应保证所有的机器人避开所有的障碍物。有约束方程：

$$\mathbf{G}_2 = \begin{bmatrix} g_{12} \\ \vdots \\ g_{r2} \end{bmatrix} \quad (10-17)$$

式中, g_{r2} 为式(10-7)中的 g_2 。

(3) 会合静止目标。所有机器人在运动终点的位置和速度应与其轨迹的终了条件相对应。对于会合静止目标的飞行轨迹将限制终点的位置为会合点的位置, 而速度为零。因此有如下约束方程：

$$\mathbf{G}_3 = \begin{bmatrix} g_{13} \\ \vdots \\ g_{r3} \end{bmatrix} \quad (10-18)$$

式中, g_{r3} 为式(10-8)中的 g_3 。

(4) 机器人之间碰撞检测。多个 FFSR 在飞行过程中, 每个机器人应避免与其他机器人相撞, 即每个机器人应以其他机器人为障碍物。约束方程应保证在运动过程中, 机器人与机器人之间的距离大于零。本章中假设机器人均为近似圆形, 可以减少碰撞检测的次数。要保证不同轨迹上同时刻的分割点的距离大于两个机器人半径之和, 则避障约束方程可以表示为：

$$\mathbf{G}_4 = k_c \cdot (r_i + r_k) - \sqrt{(x_{ij} - x_{kj})^2 + (y_{ij} - y_{kj})^2 + (z_{ij} - z_{kj})^2} \quad (10-19)$$

式中, (x_{ij}, y_{ij}, z_{ij}) 是第 i 个机器人第 j 个点的坐标; (x_{kj}, y_{kj}, z_{kj}) 是第 k 个机器人第 j 个点的坐标; r_i 是第 i 个机器人的半径; r_k 是第 k 个机器人的半径, $k > i$; k_c 是避障安全因数, $k_c \geq 1$ 。

10.2 参数优化方法

有约束条件参数优化的目的就是在满足约束条件的范围内寻找参数向量各元素的最优值以使目标函数达到最大值(或最小值)。对于有约束条件的极值问题, 通常可以用惩罚函数把它转化为无约束条件的极值问题, 然后求解, 但这种方法效率不高。本章所采用的优化方法是 Matlab 优化工具箱中的有约束问题求解最优值的算法。

Matlab 是一种面向科学和工程计算的可视化编程语言, 具有极高的编程效率。其优化工具箱(Optimization Toolbox)涉及函数的最大化和最小化问题, 它由一些对非线性函数求解最大化和最小化问题的函数和解决诸如线性规划等标准矩阵问题的函数组成。限定条件下的最小、最大、目标法和半无穷优化等问题, 所采用的原理算法是二次规划法。使用二次规划算法, 非线性条件的极值问题通常比无约束条件极值问题用更少的迭代来解决。

实质上, 该算法就是重复求解一个有约束的二次规划问题。二次规划算法公式为：

$$\min: \frac{1}{2}x^T Hx + G^T x \quad Ax \leq b \quad (10-20)$$

式中, G 是斜率, 即目标函数 f 对参数向量 x 的一阶导数; H 是矩阵, 即目标函数 f 对参数向量 x 的二阶导数。 G 是通过有限微分方法计算的, 此时记录下 x 的各个向量元素对于 f 的变化。 H 取决于每次迭代时 G 的变化, 是通过矩阵更新方法来计算的。优化中的不等式约束是关于 x 当前值约束条件的简单的线性组合。变量 b 是 x 当前值的约束评价。 A 是在当前操作点 b 对 x 的一阶导数, 可以应用有限微分方法进行数值计算。

为优化目标函数提供 x 个元素的初始值, 优化算法返回最优解。初始值必须满足所有的约束条件。返回的最优值使目标函数最大化, 并使所有的约束方程小于或等于零。如果初值得当, 该最优值是全局的最优值。

10.3 计算机仿真

为了验证本章提出的飞行轨迹优化算法的正确性, 针对在二维平面内做自由飞行运动的 FFSR 系统, 进行了数字仿真。所考虑的 FFSR 是本课题组正在研制的 FFSR 地面实验模型, 它是由机器人本体和两个机械手组成的平面型飞行机器人系统。它在 8 个气体推进器的作用下, 可以在花岗石支持平台上的二维平面内自由运动。气体推进器的分布如图 10-5 所示。假设机器人的质量为 20kg, 每个方向上可以获得的最大驱动力为 0.5N。

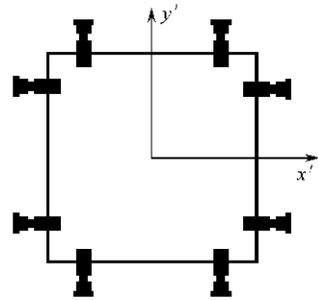


图 10-5 气体推进器分布

10.3.1 单 FFSR 接近静止目标飞行轨迹优化仿真

FFSR 接近静止目标的飞行轨迹是机器人本体在飞行过程中所经历的一条曲线, 机器人以一定的初速度开始运动, 保证机器人避开所有的障碍物, 并在会合点处, 使目标处于机器人工作空间范围内, 此时机器人的速度和加速度均为零。

假设 FFSR 从点 $(0,0)$ 飞至点 $(3,3)$, 机器人看做一个质点。障碍物的中心为 $(1.1, 1.4)$, 半径为 0.5m, 避障安全因数为 1.3。把整个轨迹分成 10 段, 设各段运动时间相等。图 10-6 所示为时间因子 $c = 5$ 时的优化轨迹, 此时时间因素的权重较大, 即要求飞行时间尽可能的短, 对于燃料消耗则做次要的要求。可以看出飞行轨迹比较接近障碍物, 飞行整个轨迹所用的运动时间为 24.063s, 燃料消耗量为 19.163N·s。图 10-7 所示为时间因子 $c = 0.5$ 时的优化轨迹, 此时燃料因数的权重较大, 即要求燃料消耗尽可能的少, 对于飞行时间则做次要的要求。可以看出飞行轨迹比较远离障碍物, 飞行整个轨迹所用的运动时间为 29.839s, 燃料消耗量为 11.706N·s。

从仿真结果看出, 通过调整时间因子可以得到不同条件下的优化轨迹, 以满足对时间和燃料权重的不同要求。通过附加的约束条件可以很好地实现机器人的避障, 通过选择避障安全因数可以调整轨迹与障碍物之间的距离。安全因数越大, 避障效果越好, 但优化的结果较差, 因此可根据具体情况而定。

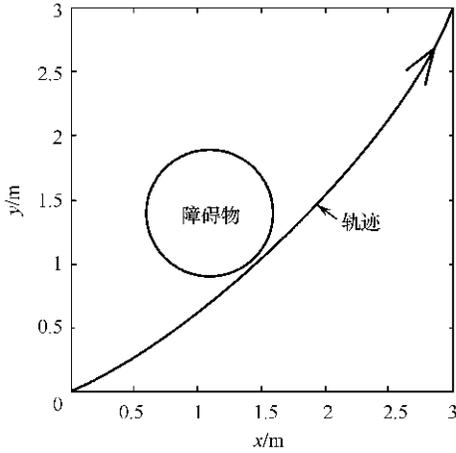


图 10-6 $c = 5$ 时的优化轨迹

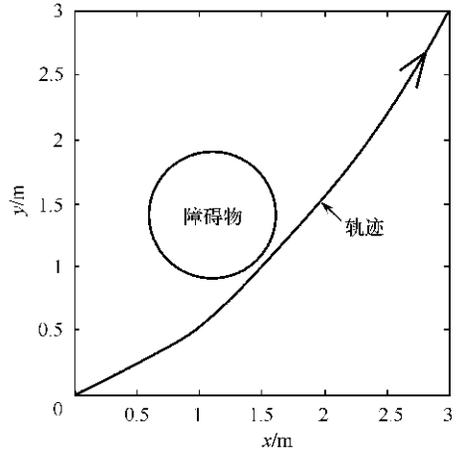


图 10-7 $c = 0.5$ 时的优化轨迹

10.3.2 单 FFSR 截击运动目标飞行轨迹优化仿真

FFSR 截击目标的飞行轨迹, 并保证在会合点处, 目标处于机器人的工作空间范围内, 并且应该满足轨迹的生成条件。

假设机器人的半径为 0.2m, 机器人本体中心到工作空间中心的距离为 0.3m。机器人从点(0.5, 0.5) 开始运动。障碍物的中心为(1, 2.7), 其半径为 1m, 选择避障安全因数为 1.1。目标从点(5.0, 4.5) 开始沿 x 轴负方向, 以 0.11m/s 的速度匀速运动。机器人和目标的运动范围是由点(0.0, 0.0)、(5.0, 0.0)、(5.0, 5.0)、(0.0, 5.0) 组成的正方形平面内。假设机器人和目标同时开始运动。把机器人整个飞行轨迹分成 10 段, 设各段运动时间相等。图 10-8 所示为时间因子 $c = 5$ 时截击运动目标的优化轨迹, 此时时间因素的权重较大, 即要求飞行时间尽可能的短, 对于燃料消耗则做次要的要求。整个轨迹所用的运动时间为 26.234s, 燃料消耗量为 23.4758N · s。图 10-9 所示为时间因子 $c = 0.5$ 时截击运动目标的优化轨迹, 此时燃料因素的权重较大, 即要求燃料消耗尽可能的少, 对于飞行时间则做次要的要求。整个轨迹所用的运动时间为 31.397s, 燃料消耗量为 14.1299N · s。

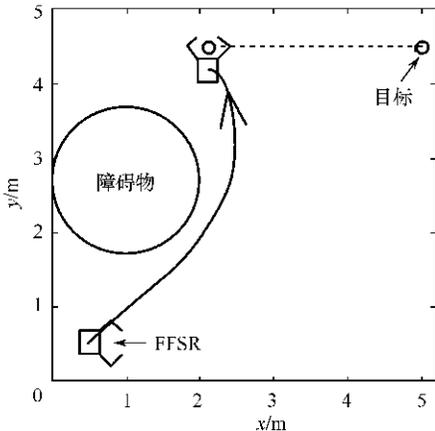


图 10-8 $c = 5$ 时的优化轨迹

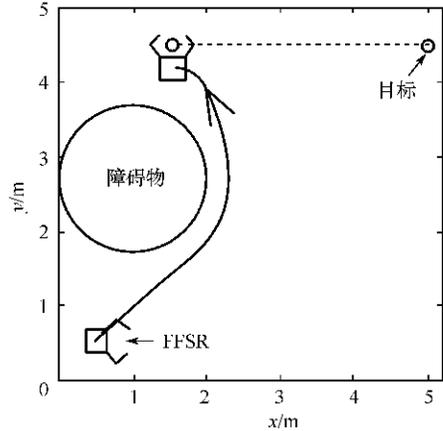


图 10-9 $c = 0.5$ 时的优化轨迹

从仿真结果可以看出,通过附加截击时刻机器人本体速度和目标运动速度的约束方程,可以保障在会合点处目标处于机器人工作空间范围内,并且机器人与目标以相同的速度运动,从而满足了截击大质量目标的轨迹生成条件。

10.3.3 多 FFSR 飞行轨迹优化仿真

多 FFSR 飞行最优轨迹不是单 FFSR 的优化轨迹的简单组合,而是整体的最优,并且机器人之间互为障碍物。

假设两个机器人同时从初始点(0.0,0.0)出发,同时到达各自的会合点(2.5,3.0)和(3.0,2.5)。机器人的半径均为 0.06m。障碍物的中心为(1.1,1.4),半径为 0.6m,避障安全因数为 1.3。把整个轨迹分成 10 段,设各段运动时间相等。图 10-10 所示为时间因子 $c = 5$ 时的优化轨迹,此时时间因素的权重较大,可以看出飞行轨迹比较接近障碍物,飞行整个轨迹每个机器人所用的运动时间为 25.904s,燃料消耗量之和为 38.881N·s。图 10-11 所示为时间因子 $c = 0.5$ 时的优化轨迹,此时燃料因数的权重较大,可以看出飞行轨迹比较远离障碍物,飞行整个轨迹每个机器人所用的运动时间为 32.050s,燃料消耗量之和为 21.991N·s。

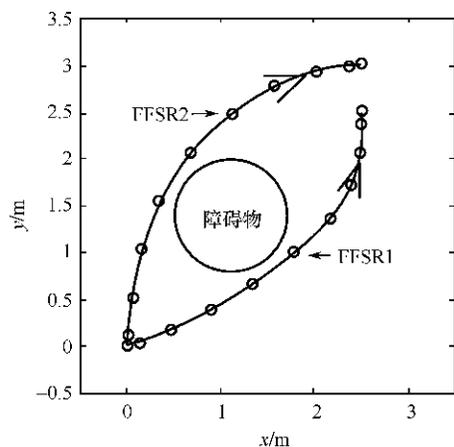


图 10-10 $c = 5$ 时的优化轨迹

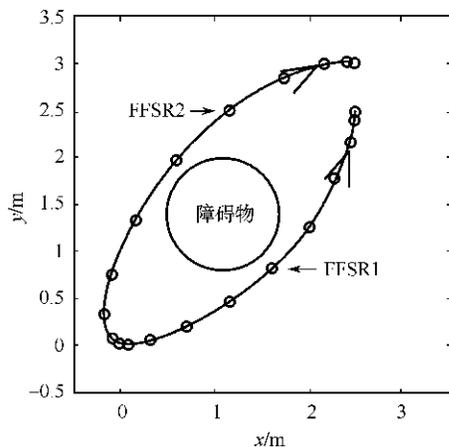


图 10-11 $c = 0.5$ 时的优化轨迹

从仿真结果可以看出,多 FFSR 飞行轨迹优化方法可以很好地优化出多 FFSR 的飞行轨迹,在会合点处目标处于所有机器人的工作空间范围内。通过附加机器人之间碰撞检测的约束条件可以实现机器人之间的碰撞检测。如果没有此约束条件得到的是基本重合的两条轨迹,机器人按该轨迹运动机器人之间将发生碰撞。

10.4 小 结

针对 FFSR 的本体飞行轨迹问题,本章提出了关于 FFSR 飞行轨迹优化算法。以飞行时间短和燃料消耗少为优化目标建立多目标优化模型,建立了目标函数,并选择优化参数和约束条件。主要建立接近静止目标轨迹优化、截击运动目标轨迹优化和多个 FFSR 飞行轨迹优化模型。本章所采用的优化方法是 MATLAB 优化工具箱中的有约束问题求解最

优值的算法。并完成了三种仿真实验。

仿真结果表明,本章提出的关于 FFSR 飞行轨迹优化算法是正确的和可行的。通过该算法可以得到 FFSR 飞行运动的最佳轨迹。通过调整燃料和时间权重——时间因子的数值,可以得到不同条件下的优化轨迹。该优化算法通过附加约束条件可以较好地实现 FFSR 的避障。它为 FFSR 实时轨迹规划奠定了基础。

第十一章 多个 FFSR 协调操作控制算法

由于空间作业的特殊性和复杂性,单个 FFSR 已难以完成复杂的任务。而多个 FFSR 具有更高的灵活性和可靠性,并具有协调作业的能力。在太空中的某些任务需要两个或多个 FFSR 作为一个小组协调操作来完成。本章主要研究多个 FFSR 协调操作同一个较大目标时的控制方法,推导出多个 FFSR 协调操作的广义雅可比矩阵,并建立 FFSR 协调操作动力学模型,推导出动力学方程,其次给出 FFSR 协调操作动力学控制方案,并证明了该控制方案的稳定性。最后用计算机仿真验证了所提出的动力学控制方法的正确性。

11.1 多 FFSR 运动学模型

目前,许多文献研究了 FFSR 完成作业任务时的运动规划问题,但是多数是针对单个基座的单臂、双臂或多臂 FFSR,是单个机器人机械臂运动的协调。而对多个 FFSR 协调操作的研究比较少。本章重点研究多个 FFSR 协调操作的运动学模型,在考虑机械臂、本体和目标之间相互干扰的基础上,分析目标上关键点的运动速度和各个机器人关节角速度的关系,进而推导出多个 FFSR 协调操作的广义雅可比矩阵。

11.1.1 运动学模型

当 FFSR 飞至交会点,此时目标处于其工作空间范围内,为了节省燃料,关闭气体推进系统,FFSR 在自由浮游的状态下捕捉和操作目标。本章研究的多个 FFSR 捕捉同一个目标之后的协调操作系统如图 11-1 所示。其符号约定如下。

ΣW : 惯性坐标系;

$\Sigma 0$: 目标坐标系;

$\Sigma j, i$: 固定于 j 机器人 i 连杆上的坐标系,定义为 $({}^i x_j, {}^i y_j, {}^i z_j)$,其中 ${}^i z_j$ 与 j 机器人的 i 关节轴方向一致;

m_0 : 目标的质量;

I_0 : 目标的转动惯量;

v_0 : 目标质心的线速度;

ω_0 : 目标质心的角速度;

m_j^i : j 机器人的 i 连杆的质量;

I_j^i : j 机器人的 i 连杆的转动惯量;

v_{jc}^i : j 机器人的 i 连杆质心的线速度;

ω_{jc}^i : j 机器人的 i 连杆质心的角速度;

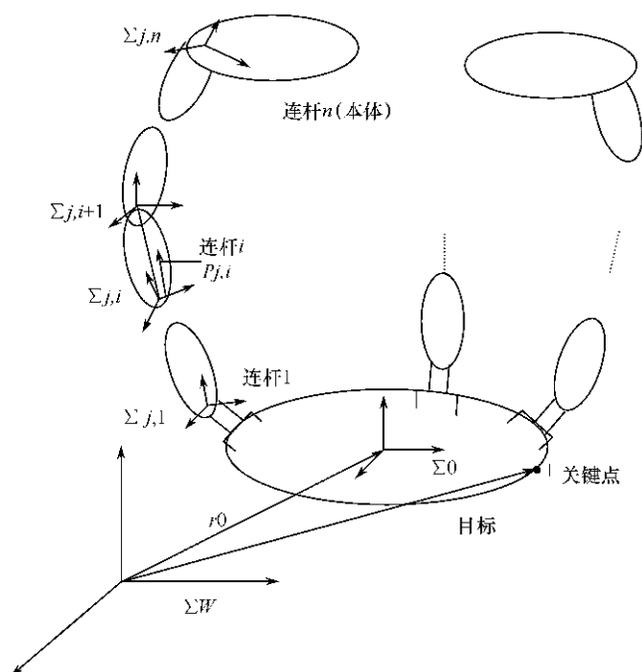


图 11-1 多个 FFSR 协调操作系统模型

r_0 : 惯性坐标系原点指向目标质心的向量;

r_{jc}^i : 惯性坐标系原点指向 j 机器人上 i 连杆质心的向量。

上述所有的向量都是在目标系中表示的。下面给出本书用到的算符, 记向量 $r = (x, y, z)^T$, 向量算符 \tilde{r} 表示为:

$$\tilde{r} = \begin{bmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{bmatrix}$$

为了简化数学模型, 本书用到以下假设。

- (1) 整个系统的所有构件均为刚体;
- (2) 整个系统不受任何外力和力矩作用, 系统的线动量和角动量守恒;
- (3) 系统的初始线动量和角动量均为零;
- (4) 各机器人均牢固地抓住目标, 末端效应器与目标之间无相对运动;
- (5) 各机器人所有的关节均为转动关节。

11.1.2 运动学方程

整个系统的线动量和角动量表达式为:

$$\mathbf{K} = m_0 \mathbf{v}_0 + \sum_{j=1}^r \sum_{i=1}^n m_j^i \mathbf{v}_{jc}^i \quad (11-1)$$

$$\mathbf{L} = I_0 \boldsymbol{\omega}_0 + \mathbf{r}_0 \times m_0 \mathbf{v}_0 + \sum_{j=1}^r \sum_{i=1}^n (I_j^i \boldsymbol{\omega}_{jc}^i + \mathbf{r}_{jc}^i \times m_j^i \mathbf{v}_{jc}^i) \quad (11-2)$$

式中, \mathbf{K} 和 \mathbf{L} 分别是系统的线动量和角动量。

j 机器人上 i 连杆质心的线速度和角速度为:

$$\begin{bmatrix} {}^i\mathbf{v}_{jc} \\ {}^i\boldsymbol{\omega}_{jc} \end{bmatrix} = \begin{bmatrix} \mathbf{E} & -{}^i\tilde{\mathbf{p}}_j \\ 0 & \mathbf{E} \end{bmatrix} \begin{bmatrix} \mathbf{v}_j \\ {}^i\boldsymbol{\omega}_j \end{bmatrix} \quad (11-3)$$

式中, ${}^i\mathbf{v}_j$ 和 ${}^i\boldsymbol{\omega}_j$ 、 ${}^i\mathbf{v}_{jc}$ 和 ${}^i\boldsymbol{\omega}_{jc}$ 分别为在 j 机器人上 i 连杆坐标系中的 j 机器人的 i 连杆关节轴与质心的线速度和角速度; \mathbf{E} 是 3×3 的单位矩阵; ${}^i\tilde{\mathbf{p}}_j$ 是 i 连杆坐标系中从 i 连杆关节轴到质心的向量。

j 机器人的 i 连杆质心的速度在目标坐标系中的表达式为:

$$\mathbf{V}_{jc}^i = \mathbf{S}_j^{ii} \mathbf{V}_j \quad (11-4)$$

式中

$$\begin{aligned} \mathbf{V}_{jc}^i &= [\mathbf{v}_{jc}^{i\top} \quad \boldsymbol{\omega}_{jc}^{i\top}]^\top \\ {}^i\mathbf{V}_j &= [{}^i\mathbf{v}_j^\top \quad {}^i\boldsymbol{\omega}_j^\top]^\top \\ \mathbf{S}_j &= \begin{bmatrix} {}^0\mathbf{R}_j & -{}^0\mathbf{R}_j {}^i\tilde{\mathbf{p}}_j \\ 0 & {}^0\mathbf{R}_j \end{bmatrix} \end{aligned}$$

${}^0\mathbf{R}_j$ 是 j 机器人上 i 连杆坐标系相对于目标系的旋转矩阵。

i 连杆关节速度 ${}^i\mathbf{V}_j$ 可以表示为:

$${}^i\mathbf{V}_j = \mathbf{q}_j(i, i-1) {}^{i-1}\mathbf{V}_j + {}^i\mathbf{n}_j \dot{\boldsymbol{\phi}}_j^i \quad j = 1, 2, \dots, r \quad (11-5)$$

式中, $\dot{\boldsymbol{\phi}}_j^i$ 是 j 机器人上 i 关节的角速度; ${}^i\mathbf{n}_j = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1]^\top$, 且

$$\mathbf{q}_j(i, i-1) = \begin{bmatrix} {}^i\mathbf{R}_j & -{}^i\mathbf{R}_j {}^{i-1}\tilde{\mathbf{l}}_j \\ 0 & {}^{i-1}\mathbf{R}_j \end{bmatrix}$$

式中, ${}^i\mathbf{R}_j$ 是 $i-1$ 连杆坐标系相对于 i 连杆坐标系的旋转矩阵; ${}^{i-1}\tilde{\mathbf{l}}_j$ 是 $i-1$ 连杆坐标系原点到 i 连杆坐标系原点的向量在 $i-1$ 连杆坐标系中的表示。 $\mathbf{q}_j(i, k)$ 具有如下特性: $\mathbf{q}_j(m, s)\mathbf{q}_j(s, p) = \mathbf{q}_j(m, p)$, $\forall m, s, p \subset n$, 且 $\mathbf{q}_j(m, m) = \mathbf{E}$, \mathbf{E} 是单位矩阵。

i 连杆关节轴和质心的速度可以表示为:

$$\mathbf{V}_j = \mathbf{Q}_j \mathbf{N}_j \dot{\boldsymbol{\phi}}_j + \mathbf{D}_j \mathbf{V}_0 \quad j = 1, 2, \dots, r \quad (11-6)$$

$$\mathbf{V}_{jc} = \mathbf{S}_j \mathbf{V}_j \quad j = 1, 2, \dots, r \quad (11-7)$$

式中

$$\begin{aligned} \mathbf{V}_j &= [{}^1\mathbf{V}_j^\top \quad \dots \quad {}^n\mathbf{V}_j^\top]^\top \\ \mathbf{V}_{jc} &= [{}^1\mathbf{V}_{jc}^\top \quad \dots \quad {}^n\mathbf{V}_{jc}^\top]^\top \\ \mathbf{V}_0 &= [{}^1\mathbf{v}_0^\top \quad \boldsymbol{\omega}_0^\top]^\top \\ \dot{\boldsymbol{\phi}}_j &= [\dot{\boldsymbol{\phi}}_j^1 \quad \dots \quad \dot{\boldsymbol{\phi}}_j^n]^\top \end{aligned}$$

$$\begin{aligned}
\mathbf{S}_j &= \text{diag}(\mathbf{S}_j^1 \quad \cdots \quad \mathbf{S}_j^n) \\
\mathbf{D}_j &= [\mathbf{q}_j^T(1,0) \quad \cdots \quad \mathbf{q}_j^T(n,0)]^T \\
\mathbf{N}_j &= \text{diag}({}^1\mathbf{n}_j \quad \cdots \quad {}^n\mathbf{n}_j) \\
\mathbf{Q}_j &= \begin{bmatrix} q_j(1,1) & 0 & \cdots & 0 \\ q_j(2,1) & q_j(2,2) & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ q_j(n,1) & q_j(n,2) & \cdots & q_j(n,n) \end{bmatrix}
\end{aligned}$$

应用式(11-7)和 \mathbf{V}_{jc} 的定义,可以得到整个系统的线动量和角动量表达式:

$$\begin{bmatrix} \mathbf{K} \\ \mathbf{L} \end{bmatrix} = (\mathbf{M}_0 + \mathbf{M}_m \mathbf{S} \mathbf{D}) \mathbf{V}_0 + \mathbf{M}_m \mathbf{S} \mathbf{Q} \mathbf{N} \dot{\boldsymbol{\phi}} \quad (11-8)$$

式中

$$\begin{aligned}
\dot{\boldsymbol{\phi}} &= [\dot{\boldsymbol{\phi}}_1^T \quad \cdots \quad \dot{\boldsymbol{\phi}}_n^T]^T \\
\mathbf{D} &= [\mathbf{D}_1^T \quad \cdots \quad \mathbf{D}_r^T]^T \\
\mathbf{S} &= \text{diag}(\mathbf{S}_1 \quad \cdots \quad \mathbf{S}_r) \\
\mathbf{Q} &= \text{diag}(\mathbf{Q}_1 \quad \cdots \quad \mathbf{Q}_r) \\
\mathbf{N} &= \text{diag}(\mathbf{N}_1 \quad \cdots \quad \mathbf{N}_r) \\
\mathbf{M}_m &= [\mathbf{M}_{m1} \quad \cdots \quad \mathbf{M}_{mr}]
\end{aligned}$$

$$\mathbf{M}_0 = \begin{bmatrix} m_0 \mathbf{E} & 0 \\ m_0 \tilde{\mathbf{r}}_0 & \mathbf{I}_0 \end{bmatrix}$$

此处

$$\mathbf{M}_{mj} = \begin{bmatrix} [m_j^1 \mathbf{E} & 0] \\ [m_j^1 \tilde{\mathbf{r}}_{jc}^1 & \mathbf{I}_j^1] \end{bmatrix} \quad \cdots \quad \begin{bmatrix} [m_j^n \mathbf{E} & 0] \\ [m_j^n \tilde{\mathbf{r}}_{jc}^n & \mathbf{I}_j^n] \end{bmatrix}$$

\mathbf{E} 是 3×3 的单位矩阵。

在目标上选取一个关键点 P , P 的速度与目标质心的速度有如下关系:

$$\mathbf{V}_P = \mathbf{J}_P \mathbf{V}_0 \quad (11-9)$$

式中, \mathbf{J}_P 是从本体质心速度向关键点 P 的速度的转换矩阵。

由前述假设(2)和(3)可知,系统的线动量和角动量守恒,并且线动量和角动量均为零。从式(11-8)可得到关节速度与目标质心速度之间的关系:

$$\mathbf{V}_0 = -\mathbf{I}_s^{-1} \mathbf{I}_m \dot{\boldsymbol{\phi}} = \mathbf{J}_s \dot{\boldsymbol{\phi}} \quad (11-10)$$

式中

$$\mathbf{I}_s = \mathbf{M}_0 + \mathbf{M}_m \mathbf{S} \mathbf{D}$$

$$\mathbf{I}_m = \mathbf{M}_m \mathbf{S} \mathbf{Q} \mathbf{N}$$

把式(11-10)代入式(11-9)得:

$$\mathbf{V}_P = -\mathbf{J}_P \mathbf{J}_s \dot{\boldsymbol{\phi}} = \mathbf{J}_e \dot{\boldsymbol{\phi}} \quad (11-11)$$

式中, $\mathbf{J}_e = -\mathbf{J}_P \mathbf{J}_s$ 定义为多个 FFSR 协调操作广义雅可比矩阵,反映了在空间微重力环境

下目标上关键点速度与所有机器人关节角速度关系。关键点速度 V_p 取决于所有机器人的关节角速度和系统的惯性参数。

11.2 多 FFSR 动力学模型

为了设计多个 FFSR 协调操作系统的控制器,推导其动力学模型是十分必要的。机器人是主动机械装置。一般情况下,它的每个自由度都是由一个单独的执行机构驱动。从控制观点来看,机器人代表了多变量的非线性的自动控制系统,每个控制任务本身就是一个动力学问题。因此,研究其动力学问题就是为了进一步讨论其控制问题。

11.2.1 关节力递推式

对于如图 11-1 所示的 FFSR 系统,作用于 j 机器人 i 连杆上的力(力和力矩)的递推式可以表示为:

$${}^i f_j = q_j(i, i-1)({}^{i-1} f_j - {}^{i-1} M_j {}^{i-1} \dot{V}_j - {}^{i-1} b_j) \quad j = 1, 2, \dots, r \quad (11-12)$$

式中, ${}^{i-1} \dot{V}_j$ 是 j 机器人 $i-1$ 连杆的加速度向量; ${}^i f_j$ 和 ${}^{i-1} f_j$ 分别为作用于 j 机器人 i 连杆和 $i-1$ 连杆上的力(力和力矩),且

$${}^i b_j = \begin{bmatrix} m_j^{ii} \tilde{\omega}_j^i \tilde{\omega}_j^i p_j \\ \tilde{\omega}_j^i I_j^i \tilde{\omega}_j^i \end{bmatrix}$$

$${}^i M_j = \begin{bmatrix} m_j^i E & -m_j^i \tilde{p}_j \\ m_j^{ii} \tilde{p}_j & I_j^i \end{bmatrix}$$

$\tilde{\omega}_j^i$ 是 $\tilde{\omega}_j$ 的操作因子,其他变量在式(11-3)、式(11-4)和式(11-5)均已定义。假设 0 连杆即末端效应器非常小,忽略其质量和转动惯量,则应用式(11-12)的力递推式,可以得到如下力式:

$${}^i f_j = q_j(i, 0) {}^0 f_j - \sum_{k=1}^{i-1} q_j(i, k) {}^k M_j {}^k \dot{V}_j - \sum_{k=1}^{i-1} q_j(i, k) {}^k b_j \quad (11-13)$$

式中, ${}^0 f_j$ 是目标作用于 j 机器人末端效应器上的力在目标坐标系中的表示。式(11-13)描述了 j 机器人所有 n 个连杆的力和加速度之间的关系,因此 j 机器人的 n 个力式可以写成:

$$f_j = -A_j(M_j \dot{V}_j + b_j) + D_j {}^0 f_j \quad j = 1, 2, \dots, r \quad (11-14)$$

式中

$$f_j = [{}^1 f_j^T \quad \dots \quad {}^n f_j^T]^T$$

$$\dot{V}_j = [{}^1 \dot{V}_j^T \quad \dots \quad {}^n \dot{V}_j^T]^T$$

$$b_j = [{}^1 b_j^T \quad \dots \quad {}^n b_j^T]^T$$

$$D_j = [q_j^T(1, 0) \quad \dots \quad q_j^T(n, 0)]^T$$

$$M_j = \text{diag}({}^1 M_j \dots {}^n M_j)$$

$$\mathbf{A}_j = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ q_j(2,1) & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ q_j(n,1) & q_j(n,2) & \cdots & 0 \end{bmatrix}$$

对于 r 个机器人,加速度和力之间的关系可以表示为:

$$\mathbf{A}(\mathbf{M}_q \dot{\mathbf{V}} + \mathbf{b}) = \mathbf{D}\mathbf{f}^0 - \mathbf{f} \quad (11-15)$$

式中

$$\begin{aligned} \dot{\mathbf{V}} &= [\dot{\mathbf{V}}_1^T \quad \cdots \quad \dot{\mathbf{V}}_r^T]^T \\ \mathbf{f} &= [\mathbf{f}_1^T \quad \cdots \quad \mathbf{f}_r^T]^T \\ \mathbf{f}^0 &= [{}^0\mathbf{f}_1^T \quad \cdots \quad {}^0\mathbf{f}_r^T]^T \\ \mathbf{b} &= [\mathbf{b}_1^T \quad \cdots \quad \mathbf{b}_r^T]^T \\ \mathbf{D} &= \text{diag}(\mathbf{D}_1 \quad \cdots \quad \mathbf{D}_r) \\ \mathbf{M}_q &= \text{diag}(\mathbf{M}_1 \quad \cdots \quad \mathbf{M}_r) \\ \mathbf{A} &= \text{diag}(\mathbf{A}_1 \quad \cdots \quad \mathbf{A}_r) \end{aligned}$$

11.2.2 动力学方程

由于整个系统不受外力作用,因此 n 连杆(飞行器)的力平衡方程可以表示为:

$${}^n\mathbf{f}_j - {}^n\mathbf{M}_j \dot{\mathbf{V}}_j - {}^n\mathbf{b}_j = 0 \quad (11-16)$$

式中, ${}^n\mathbf{M}_j$ 是 j 机器人上 n 连杆的惯性矩阵; ${}^n\mathbf{f}_j$ 是作用于 j 机器人 n 连杆上的力向量。整个系统有 r 个机器人,所以共有 r 个力平衡方程,其统一的表达式为:

$$\mathbf{f}^n - \mathbf{M}_n \dot{\mathbf{V}}_n - \mathbf{b}_n = 0 \quad (11-17)$$

式中

$$\begin{aligned} \mathbf{f}^n &= [{}^n\mathbf{f}_1^T \quad \cdots \quad {}^n\mathbf{f}_r^T]^T \\ \dot{\mathbf{V}}_n &= [{}^n\dot{\mathbf{V}}_1^T \quad \cdots \quad {}^n\dot{\mathbf{V}}_r^T]^T \\ \mathbf{b}_n &= [{}^n\mathbf{b}_1^T \quad \cdots \quad {}^n\mathbf{b}_r^T]^T \\ \mathbf{M}_n &= \text{diag}({}^n\mathbf{M}_1 \quad \cdots \quad {}^n\mathbf{M}_r) \end{aligned}$$

${}^n\mathbf{f}_j$ 可由式(11-14)表示的向量方程的第 n 个方程表达:

$${}^n\mathbf{f}_j = \mathbf{U}_j {}^0\mathbf{f}_j - \mathbf{L}_j(\mathbf{M}_j \dot{\mathbf{V}} + \mathbf{b}_j) \quad (11-18)$$

式中

$$\begin{aligned} \mathbf{U}_j &= \mathbf{q}_j(n,0) \\ \mathbf{L}_j &= [\mathbf{q}_j(n,1) \quad \mathbf{q}_j(n,2) \quad \cdots \quad \mathbf{q}_j(n,n-1) \quad 0] \end{aligned}$$

对于整个系统, r 个机器人 n 连杆的作用力可以表示为:

$$\mathbf{f}^n = \mathbf{U}\mathbf{f}^0 - \mathbf{L}(\mathbf{M}_q \dot{\mathbf{V}} + \mathbf{b}) \quad (11-19)$$

式中

$$\begin{aligned} \mathbf{U} &= \text{diag}(\mathbf{U}_1 \quad \cdots \quad \mathbf{U}_r) \\ \mathbf{L} &= \text{diag}(\mathbf{L}_1 \quad \cdots \quad \mathbf{L}_r) \end{aligned}$$

应用式(11-17)和式(11-18), n 连杆的加速度可以表示为:

$$\dot{\mathbf{V}}_n = \mathbf{M}_n^{-1} [\mathbf{U}f^0 - \mathbf{L}(\mathbf{M}_q \dot{\mathbf{V}} + \mathbf{b}) - \mathbf{b}_n] \quad (11-20)$$

应用运动学关系式, j 机器人 n 连杆的速度可以用目标的速度和 j 机器人关节速度表示为:

$${}^n \mathbf{V}_j = \mathbf{H}_j \mathbf{V}_0 + \mathbf{G}_j \mathbf{N}_j \dot{\boldsymbol{\phi}}_j \quad (11-21)$$

式中

$$\mathbf{H}_j = \mathbf{q}_j(n, 0)$$

$$\mathbf{G}_j = [\mathbf{q}_j(n, 1) \quad \cdots \quad \mathbf{q}_j(n, n)]$$

$$\mathbf{N}_j = \text{diag}({}^1 \mathbf{n}_j \quad \cdots \quad {}^n \mathbf{n}_j)$$

对于整个系统, 共有 r 个方程, 可以合并为:

$$\mathbf{V}_n = \mathbf{H} \mathbf{V}_0 + \mathbf{G} \mathbf{N} \dot{\boldsymbol{\phi}} \quad (11-22)$$

式中

$$\mathbf{H} = [\mathbf{H}_1^T \quad \cdots \quad \mathbf{H}_r^T]^T$$

$$\mathbf{G} = [\mathbf{G}_1^T \quad \cdots \quad \mathbf{G}_r^T]^T$$

$$\mathbf{N} = \text{diag}(\mathbf{N}_1 \quad \cdots \quad \mathbf{N}_r)$$

对式(11-22)两边求导, 可得 n 连杆加速度的表达式:

$$\dot{\mathbf{V}}_n = \dot{\mathbf{H}} \mathbf{V}_0 + \mathbf{H} \dot{\mathbf{V}}_0 + \dot{\mathbf{G}} \mathbf{N} \dot{\boldsymbol{\phi}} + \mathbf{G} \mathbf{N} \ddot{\boldsymbol{\phi}} \quad (11-23)$$

式中, N 是常数。对式(11-10)两边求导, 可得:

$$\dot{\mathbf{V}}_0 = \mathbf{J}_s \ddot{\boldsymbol{\phi}} + \dot{\mathbf{J}}_s \dot{\boldsymbol{\phi}} \quad (11-24)$$

把式(11-10)、式(11-20)和式(11-24)代入式(11-23), 得:

$$\dot{\mathbf{V}} = (\mathbf{L} \mathbf{M}_q)^{-1} \times [\mathbf{U}f^0 - \mathbf{L} \mathbf{b} - \mathbf{b}_n - \mathbf{M}_n (\mathbf{J}_s \dot{\mathbf{H}} + \dot{\mathbf{J}}_s \mathbf{H} + \dot{\mathbf{G}} \mathbf{N}) \dot{\boldsymbol{\phi}} - \mathbf{M}_n (\mathbf{H} \mathbf{J}_s + \mathbf{G} \mathbf{N}) \ddot{\boldsymbol{\phi}}] \quad (11-25)$$

应用式(11-5)中 ${}^i \mathbf{n}_j$ 的定义, j 机器人 i 关节的力矩可以定义为:

$${}^i \boldsymbol{\tau}_j = {}^i \mathbf{n}_j^T \mathbf{f}_j \quad (11-26)$$

并应用关节力矩的定义和式(11-15)和式(11-25), 可以得到在关节空间中, 多个 FFSR 协调操作系统的动力学方程为:

$$\mathbf{M} \ddot{\boldsymbol{\phi}} + \mathbf{C} = \boldsymbol{\tau} - \mathbf{J}^T \mathbf{f}^e \quad (11-27)$$

式中

$$\boldsymbol{\tau} = [{}^1 \boldsymbol{\tau}_1 \quad \cdots \quad {}^n \boldsymbol{\tau}_1 \quad \cdots \quad {}^1 \boldsymbol{\tau}_r \quad \cdots \quad {}^n \boldsymbol{\tau}_r]^T$$

$$\mathbf{M} = \mathbf{N}^T \mathbf{A} \mathbf{L}^{-1} \mathbf{M}_n (\mathbf{H} \mathbf{J}_s + \mathbf{G} \mathbf{N})$$

$$\mathbf{C} = \mathbf{N}^T \mathbf{A} [(\mathbf{E} - \mathbf{A}) \mathbf{b} + \mathbf{L}^{-1} (\mathbf{b}_n + \mathbf{M}_n (\mathbf{J}_s \dot{\mathbf{H}} + \dot{\mathbf{J}}_s \mathbf{H} + \dot{\mathbf{G}} \mathbf{N}) \dot{\boldsymbol{\phi}})]$$

$$\mathbf{J}^T = \mathbf{N}^T (\mathbf{A} \mathbf{L}^{-1} \mathbf{U} - \mathbf{D})$$

在式(11-27)中, τ 为关节力矩向量; ϕ 为关节位置向量; M 为系统的惯性矩阵; C 为离心力和向心力向量; f^e 为所有机器人末端效应器作用于物体上的向量; J 为将末端效应器作用力转化为关节作用力的雅可比矩阵。

目标的动力学方程为:

$$M_0 \ddot{x}_0 + C_0 = W^T f^e \quad (11-28)$$

式中, M_0 为目标的惯性矩阵; C_0 为离心力和向心力向量; f^e 为所有机器人末端效应器作用于物体上的向量; $W = [E_1 \quad \cdots \quad E_r]^T$, 此处 E_j 为 6×6 的单位矩阵, $j = 1, 2, \dots, r$ 。

11.3 协调控制算法

多个 FFSR 协调操作的控制问题, 既比单个 FFSR 操作复杂得多, 又不同于多个固定机座机器人的协调。由于多个 FFSR 同时操作一个目标, 引入了复杂的约束。机器人末端效应器和目标之间弹力的存在, 使末端效应器和目标之间产生内力作用。如果不对这种力进行控制将损坏机器人或目标。因此对于多个 FFSR 协调的空间作业, 应同时控制以下两个主要变量: ① 目标的位置; ② 内力, 此力不使目标产生运动。本节在前面推导出的多个 FFSR 协调操作运动学和动力学模型的基础上, 提出同时控制目标位置和内力的控制方案。此处假设运动学和动力学模型是完全确定的。

11.3.1 协调控制方案

多个 FFSR 协调操作的控制器由两部分组成: 一部分是目标位置控制器; 另一部分是内力控制器。该控制器不使目标产生运动。这种控制方案可用下式描述:

$$\tau = \tau^p + \tau^i \quad (11-29)$$

式中, τ^p 和 τ^i 分别是目标位置和内力控制命令。

目标位置控制命令可以定义为:

$$\begin{aligned} \tau^p = & (MJ_s^{-1} + J^T W^{+T} M_0) \times [\ddot{x}_0^d + K_v^p (\dot{x}_0^d - \dot{x}_0) + K_p^p (x_0^d - x_0)] + \\ & C + J^T W^{+T} C_0 - MJ_s^{-1} \dot{J}_s \dot{\phi} \end{aligned} \quad (11-30)$$

式中, x_0^d , \dot{x}_0^d 和 \ddot{x}_0^d 分别为目标的广义位置、速度和加速度; K_p^p 和 K_v^p 分别为位置和速度误差反馈系数矩阵; $W^+ = [\alpha_1^T E_1 \quad \cdots \quad \alpha_r^T E_r]$, 此处 E_j 为 6×6 的单位矩阵, $j = 1, 2, \dots, r$, α_j 为载荷分配矩阵因子, $j = 1, 2, \dots, r$ 。该控制方案中载荷分配是由载荷分配矩阵因子决定的, 该矩阵因子满足关系式: $\alpha_1 + \cdots + \alpha_r = E$, E 为 6×6 的单位矩阵。如果 r 个机器人均相同, 则载荷分配矩阵因子可以选为: $\alpha_j = E_j^T E_j (W^T W)^{-1}$ 。

内力控制命令可以定义为内力误差的比例和积分控制:

$$\tau^i = J^T [f_d^i + K_p^i (f_d^i - f^i) + K_i^i \int (f_d^i - f^i) dt] \quad (11-31)$$

式中, f_d^i 和 f^i 分别为理想和实际的内力向量, 可以表示为 $[(f_{d1}^i)^T \quad \cdots \quad (f_{dr}^i)^T]^T$ 和

$[(f_1^i)^T \cdots (f_r^i)^T]^T$, 其中 f_{dj}^i 和 f_j^i 分别为 j 机器人末端效应器提供的理想和实际的内力, $j = 1, 2, \dots, r$, 可用下式表示:

$$f_{dj}^i = ([0 \cdots E_j \cdots 0] - E_j \alpha_j W^T) f_d^e \quad (11-32)$$

$$f_j^i = ([0 \cdots E_j \cdots 0] - E_j \alpha_j W^T) f^e \quad (11-33)$$

在式(11-31) ~ 式(11-33)中, K_p^i 和 K_i^i 分别为比例和积分内力误差反馈系数矩阵, 且 $K_p^i = \gamma_p E$ 和 $K_i^i = \gamma_i E$, γ_p 和 γ_i 是正的标量; $f_d^e = [(f_{d1}^e)^T \cdots (f_{dr}^e)^T]^T$, 式中 f_{dj}^e 为 j 机器人末端效应器作用于目标上的理想广义力, $j = 1, 2, \dots, r$; E_j 为 6×6 的单位矩阵, $j = 1, 2, \dots, r$.

从式(11-33)的内力的定义可知, 内力向量 f^i 可以表示为: $(E - W^{+T}W^T)f^e$, 并且它属于 W^T 的零空间. f^e 可以分为两个部分, 一个是 $W^{+T}W^T f^e$, 直接使目标产生运动, 另一个是 $(E - W^{+T}W^T)f^e$, 代表内力, 不使目标产生运动. 同样, f_d^e 也可以分为两部分, 即 $W^{+T}W^T f_d^e$ 和 $(E - W^{+T}W^T)f_d^e$.

11.3.2 稳定性分析

前面给出了多个 FFSR 协调操作的控制方案, 在这个控制系统中, 应用位置和速度反馈以及比例和积分的内力反馈, 同时控制目标的位置和内力. 本节将分析所给出控制方案的闭环系统的稳定性, 并且证明目标位置和内力误差异步收敛. 把式(11-27)代入式(11-28), 并应用式(11-14)和式(11-29) ~ 式(11-33), 可得到如下的闭环方程:

$$(M_0 + W^T J^{-T} M J_s^{-1}) \ddot{x}_0 = (M_0 + W^T J^{-T} M J_s^{-1}) [\ddot{x}_0^d + K_v^p (\dot{x}_0^d - \dot{x}_0) + K_p^p (x_0^d - x_0)] + W^T [f_d^i + K_p^i (f_d^i - f^i) + K_i^i \int (f_d^i - f^i) dt] \quad (11-34)$$

既然 $W^T W^{+T} = E$, 所以理想和实际的内力 f_d^i 和 f^i 属于 W^T 的零空间. 从 K_p^i 和 K_i^i 的表达式可知, 上式的右边最后一项为零. 因此式(11-34)可以写成:

$$(M_0 + W^T J^{-T} M J_s^{-1}) (\ddot{e}^p + K_v^p \dot{e}^p + K_p^p e^p) = 0 \quad (11-35)$$

式中, $e^p = x_0^d - x_0$, 为目标位置误差. 这个闭环系统的运动动态特性是由式(11-34)控制的. 如果 $M_0 + W^T J^{-T} M J_s^{-1}$ 是奇异的, 则系统的秩将减少. 本书考虑系统秩不减少时的控制器, 此时该矩阵是非奇异的. 因此, 目标位置动态误差可由下式控制:

$$(\ddot{e}^p + K_v^p \dot{e}^p + K_p^p e^p) = 0 \quad (11-36)$$

下面建立内力误差模型以证明目标位置和内力收敛. 基于载荷分配矩阵因子 α_j , $j = 1, 2, \dots, r$, 式(11-28)中的广义力 $f^e = [(f_1^e)^T \cdots (f_r^e)^T]^T$ 可以分为两部分, 用式表示如下:

$$f_j^e = E_j \alpha_j (M_0 \ddot{x}_0 + C_0) + ([0 \cdots E_j \cdots 0] - E_j \alpha_j W^T) f^e \quad (11-37)$$

上式右边的第一项代表直接使目标产生运动的力, 第二项代表不使目标产生运动的内力. 把式(11-37)代入式(11-27), 并应用式(11-29) ~ 式(11-33), 可得如下的闭环方程:

$$(W^{+T}M_0 + J^{-T}MJ_s^{-1})(\dot{e}^p + K_v^p \dot{e}^p + K_p^p \dot{e}) + (E + K_p^i)e^i + K_i^i \int e^i dt \quad (11-38)$$

式中, e^i 是内力误差, 定义为 $e^i = f_d^i - f^i$ 。

应用式(11-25)的目标运动动态误差方程, 可以得到内力误差模型:

$$e^i + K^i \int e^i dt = 0 \quad (11-39)$$

式中

$$K^i = (E + K_p^i)^{-1} K_i^i$$

11.4 计算机仿真

为了验证本章推导出的动力学模型以及提出的动力学控制方案的正确性, 针对两个单臂三自由度 FFSR 同时操作一个空间目标的任务, 在 IBM-PC 机上进行了计算机仿真, 系统模型如图 11-2 所示。仿真软件采用 Matlab 可视化编程工具。整个系统的参数如表 11-1 所列。初始条件: $r_0 = [0 \quad -1.0]^T$, $\phi_0 = 0$, $\phi_0^0 = -30^\circ$, $\phi_0^1 = 0^\circ$, $\phi_0^2 = 100^\circ$, $\phi_1^1 = 20^\circ$, $\phi_1^2 = 40^\circ$, $\phi_1^3 = 20^\circ$, $\phi_2^1 = -20^\circ$, $\phi_2^2 = -40^\circ$, $\phi_2^3 = -20^\circ$ 。选取 $\Delta t = 15\text{ms}$ 。

表 11-1 两个 FFSR 协调操作系统参数表

	长度或直径 /m	质量 /kg
连杆 1	1.0	1.0
连杆 2	1.0	1.0
连杆 3	1.0	10.0
目标 0	1.6	1.0

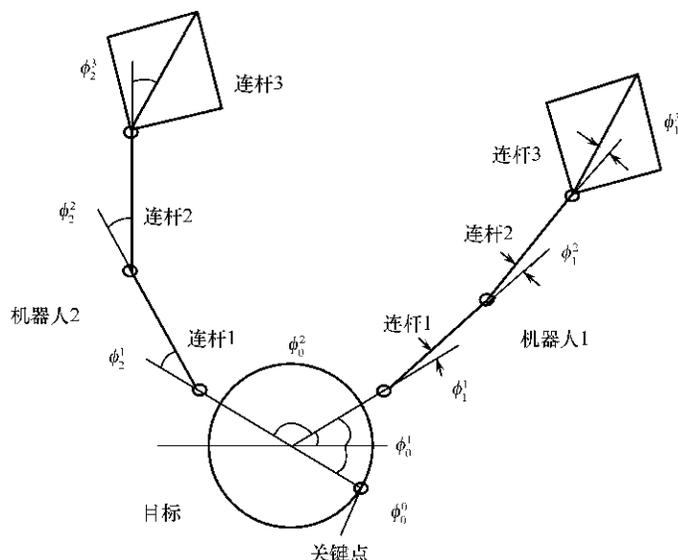


图 11-2 两个 FFSR 协调操作系统模型

两个机器人牢固地抓住目标,对目标进行操作。目标的运动可由 $x(t)$ 、 $y(t)$ 和 $\phi(t)$ 控制,其中 $\phi(t)$ 代表目标的方向。设目标的理想轨迹为:

$$x_d(t) = 0.2\cos(2t)(\text{m}) \quad y_d(t) = -1.0 + 0.2\cos(2t)(\text{m}) \quad \phi_d(t) = 0.1\cos(2t)(\text{m})$$

同时假设有三个内力 f_x^i 、 f_y^i 和 f_ϕ^i ,这些内力可由下面三个常量控制: $f_x^i = 10\text{N}$ 、 $f_y^i = 10\text{N}$ 和 $f_\phi^i = 5\text{N} \cdot \text{m}$ 。

既然两个机器人的结构完全相同,它们为目标运动和作用于目标上的内力提供相同的贡献,因此载荷分配因子可以选为: $\alpha_1 = 0.5E_3$ 和 $\alpha_2 = 0.5E_3$,其中 E_3 是 3×3 的单位矩阵。位置和速度误差反馈系数矩阵选为: $K_p^p = 70E_3$ 和 $K_v^p = 16.7E_3$,内力误差反馈系数矩阵选为: $K_p^i = 0.1E_3$ 和 $K_i^i = 1.2E_3$ 。图 11-3 给出了只有位置控制时的仿真结果,图 11-4 既有位置控制又有内力控制时的仿真结果。

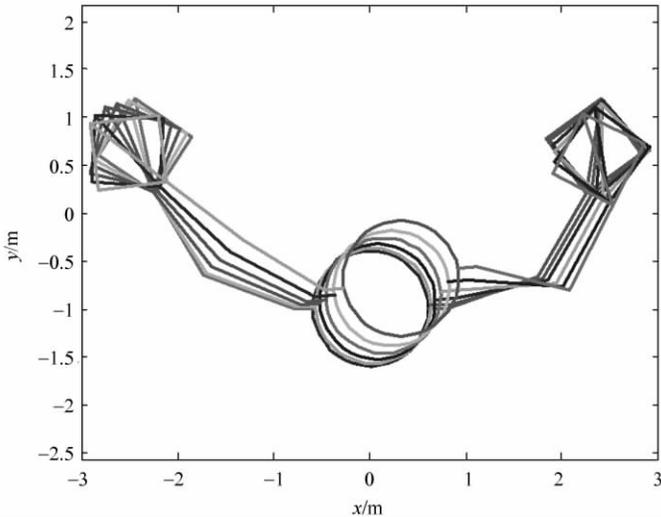


图 11-3 位置控制仿真结果

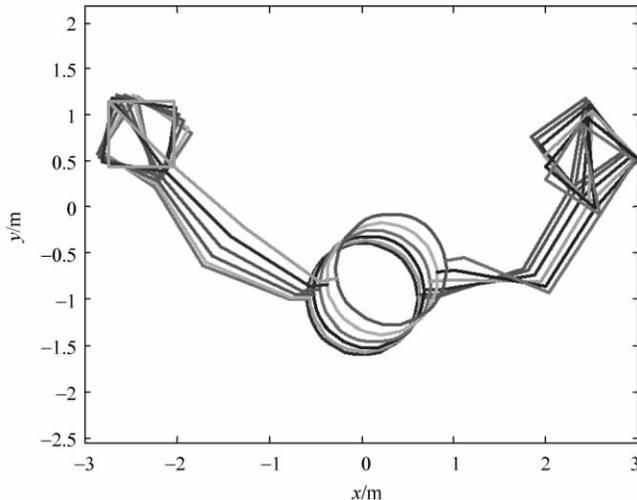


图 11-4 位置和内力控制仿真结果

11.5 小 结

针对多个 FFSR 协调操作一个较大目标的过程,本章在推导出运动学和动力学模型的基础上,提出了控制方案,同时控制目标的位置和内力,即将目标的位置和内力最终转化成关节的驱动力,从而实现力矩控制。通过仿真实验证明了本章的运动学模型以及控制方案的正确性。该控制方法在 FFSR 在空间进行装配作业时,得到广泛应用。

第十二章 FFSR 地面实验平台

FFSR 工作在太空微重力环境中,由于基座不固定,其运动学和动力学控制问题要比地面机器人复杂得多。为了验证在空间微重力环境下,FFSR 的运动特性、运动学和动力学模型以及姿态控制算法和路径规划算法的可行性和正确性,最有效的方法是在地面上模拟空间的微重力环境,并在这种环境下做实际机器人模型各种控制算法的各种仿真实验。下面介绍双臂 FFSR 地面实验平台系统及相关仿真实验。

12.1 地面实验平台系统组成

FFSR 地面实验平台系统由机器人模型、视觉系统、无线通信系统、网络系统、操作台、运动规划系统、气浮系统 7 个子模块构成。整个实验平台的系统组成如图 12-1 所示。

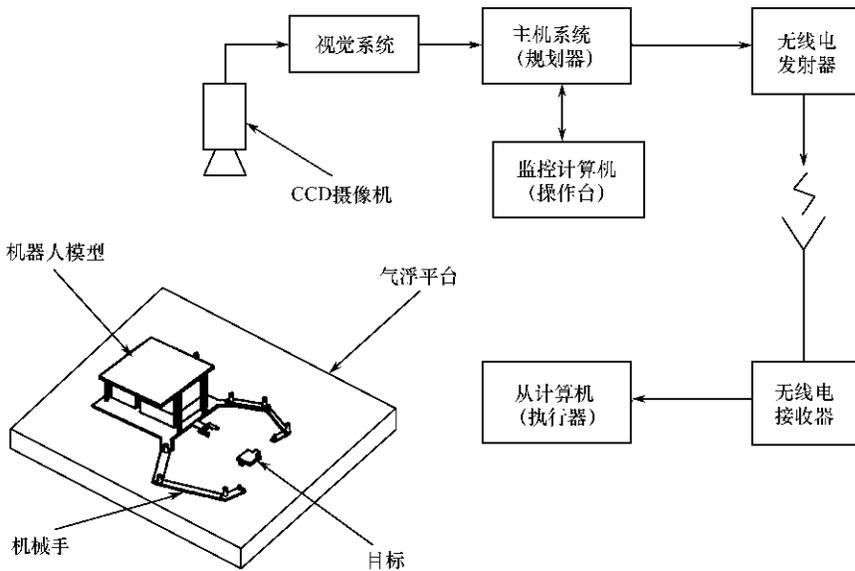


图 12-1 FFSR 地面实验平台系统

通过气浮系统提供高压空气把机器人模型在气浮平台上浮起来,以在地面上模仿二维的空间微重力环境。机器人模型完全独立,与外界没有硬件连接,从而消除了外界设备对实验效果的影响。视觉子系统为整个平台系统提供关于环境的反馈信息。在机器人模型和目标上选择了 9 个关键点,打上彩色标记,视觉子系统通过识别这 9 个点,提供 9 个点的坐标,以实现系统对环境的实时反馈。遥控操作子系统为操作员提供操作平台。它通过视觉反馈系统提供的环境信息形成三维图像,实时显示机器人模型和目标的相对位置和运

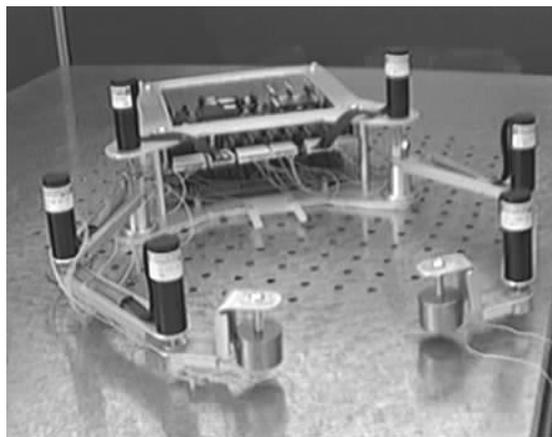
动速度。操作员可以通过键盘、鼠标等外部设备输入控制命令,操作系统把命令传给规划器,由规划器去执行。规划器主要任务是通过视觉反馈系统提供的环境信息和接收操作系统发送的命令完成任务规划和路径规划,并通过无线发射模块向机器人模型发出执行命令。机器人模型通过无线接收模块接收到规划器发来的命令后,由 CPU 把它译码,并控制电机驱动电路驱动电机运动,以完成具体系列动作。

12.2 机器人模型

机器人模型是整个地面实验平台系统的主体,它是参考航天员的基本功能设计的,它由一个本体和两个灵活操作的机械臂组成,为双臂 6 自由度平面型机器人,其实物照片如图 12-2 所示。机器人模型完全独立,和主机的联系是通过无线通信实现的,且电源也安装在机器人上。机器人模型总质量 1.6kg,为了减轻机器人的质量,机械结构均采用铝材料。机器人模型的主要参数如表 12-1 所列。



(a)



(b)

图 12-2 参考航天员的概念性机器人模型实物照片

(a) 航天员; (b) 机器人模型。

机器人模型的组成如下。

(1) 机器人本体。本体是机器人的基本结构,由电源、无线通信接收模块、主控板、关节电机驱动装置等模块组成。

(2) 机械臂。两个操作机械臂分别有三个转动关节和一个抓取装置。关节驱动电机采用瑞士微电机公司的 1516 型直流微电机,减速装置采用与电机配套的行星齿轮减速器,电机上装有编码器作为内部传感器。抓取结构采用电磁控制。

(3) 主控板。控制板由两个 CPU 和两路电机驱动电路构成。由于机器人安装 6 个电机,为了提高控制速度,故采用两个 CPU,分别控制 3 个电机。其主要功能是通过接收无线模块接收规划器发来的控制命令,并把它译码,然后驱动关节电机运动,同时控制编码器提供角度反馈信息。

(4) 电源。电源采用直流干电池,电压为 7.5V。

表 12-1 机器人模型的主要参数

模块名	参 数	模块名	参 数
① 本体		③ 关节电机	
本体尺寸	150mm × 150mm × 60mm	最大驱动力矩	0.8mN · m
质量	1.35kg	最高转数	12000r/min
② 机械手臂		④ 减速器	
自由度数	3	减速比	369 : 1
长度	22cm	⑤ 编码器	
质量	0.125kg	通道	2
抓手作用力	0.3N	分辨率	512L

12.3 主控板

控制板在整个平台中起着核心作用,其控制系统的精度决定着整个平台的性能,特别是其中的电机驱动模块的控制精度直接决定整个试验平台的可靠性。电机驱动模块的速度直接影响了实际机器人模型的实时性。

采用了专用电机控制芯片 L298 以及模糊逻辑控制算法,来简化设计和调试过程;并用单纯形法对模糊控制器中的参数进行优化,以提高控制精度。为了提高电机驱动模块的控制速度,采用双 CPU 结构,CPU 采用 CHMOS 高性能 16 位单片机 80C196KC,同时采用 Intel 公司设计的高级语言 PL/M 语言进行编程,简化了整个控制系统软件的设计过程。

12.3.1 控制板总体方案设计

1. 控制板结构

在本控制系统中,由于机器人上安装 6 个电机,为了满足系统的实时性要求,采用双 CPU 控制,每个 CPU 控制 3 个电机。控制板的组成框图如图 12-3 所示。其主要功能是控制板上 CPU 通过无线通信模块接收规划器发来的控制命令,驱动关节电机运动,CPU 根据编码器提供的反馈信息精确地控制电机。

2. 控制板的设计与实现

控制板的主要功能是接收运动规划子系统通过无线发射模块发射的命令,CPU 根据特定的协议把接收到的命令转化为驱动电机转动的 PWM 信号,通过由 L298 构成的电机驱动电路驱动电机转动给定的角度。因为这个系统具有很好的实时性,所以基于以上任务和要求,对各主要部分做如下的选择。

地面试验平台要求控制系统有很好的可靠性,所以选用商用无线通信模块 BIM-418F 作为无线电通信设备,并采用特别的编码方式使通信有很好的抗干扰能力。其中 CPU 采用 80C196KC,80C196KC 是 CHMOS 高性能 16 位单片机中的一个新分支,它有 3 个 PWM(脉宽调制)输出——这对于同时控制多个电机特别方便、一个外设事务服务器 PTS(Peripheral Transaction Server)——这大大降低了中断服务的开销、10 位 /8 位可选择的 A/D 转换器——这使得用户可以选择采用 8 位转换方式以加速转换过程或采用 10 位方式提高精度、而且它可以采用 16MHz 的晶振,另外 80C196KC 中的定时器 T2 既可以

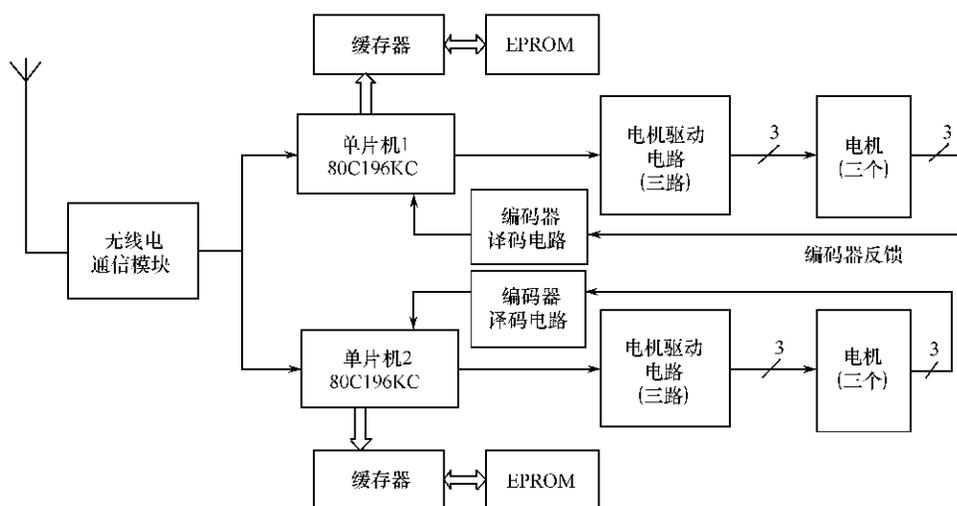


图 12-3 主控板框图

由外部提供时钟,也可以由内部提供时钟。基于 80C196KC 的以上特点,本系统中采用它作为控制板的 CPU。

为了更好地驱动直流电机,采用双 H 桥驱动器 L298。L298 是双 H 桥高电压大电流功率集成电路,可用来驱动继电器、线圈、直流电机和步进电机等感性负载,同时它还具有过热保护功能。

本系统中采用的是 512 线光学编码器,这种编码器可以有两路脉冲 A 相和 B 相输出,相差为 90° ,经过适当的译码电路可以由 A 相和 B 相译出位置计数信息(CNT)和转动方向(DIR)信息。有了 CNT 和 DIR 两个信息,就可以方便地计算出编码器的净位置增量,从而可以计算出各关节电机的转角。利用 80C196KC 的高速输入 HSI 使每 3 个电机与一个 80C196KC 接口相连。编码器译码电路的输出 CNT 反馈至一个 HSI 脚,该脚应配置成可以同时识别上升和下降沿事件。译码电路的另一输出 DIR 送至一个端口脚上,以决定方向。其工作原理如图 12-4 所示。

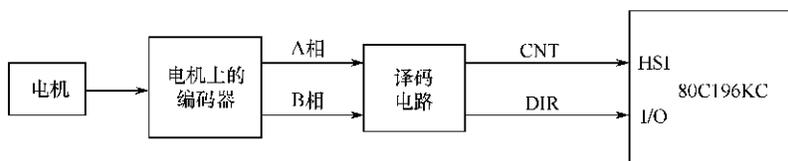


图 12-4 编码器与 HSI 的接口

12.3.2 直流电机的模糊控制算法

在本系统中,用模糊逻辑控制各关节电机的转角。模糊逻辑自从出现以来,已经应用到相当广泛的领域中。模糊系统消除了常规系统经常存在的鲁棒性与灵敏度之间的矛盾。过去一个调整精细、易于响应的系统意味着该系统对使之不稳定的作用非常敏感;反之,一个系统能抗干扰是指该系统不敏感。用模糊逻辑控制的系统就可能既具有鲁棒性以适应参数的变化,又同时非常敏感。所以,本系统采用模糊逻辑控制器控制各关节的电机。

通常将模糊控制器输入变量的个数成为模糊控制器的维数。从理论上讲,模糊控制器的维数越高,控制越精细。但是维数越高,模糊控制规则变得过于复杂,控制算法的实现相当困难。所以目前广泛采用的是二维模糊控制器。二维模糊控制器以误差和误差变化作为输入变量。本章也采用二维模糊控制器。本章采用一种改进的模糊控制器,这种控制器中采用了归一 Fuzzy 量化方法和带修正函数的 Fuzzy 规则自调整方法。

1. 归一 Fuzzy 量化

对于本系统,取系统误差和误差的变化为输入变量,系统误差为:

$$e(k) = \theta_1(k) - \theta_0(k) \quad (12-1)$$

误差的变化为:

$$ec(k) = e(k) - e(k-1) \quad (12-2)$$

式中, θ_0 为系统的给定转角; θ_1 为实际转角; $e(k)$ 和 $ec(k)$ 分别为 kT 时刻的系统误差值及其变化值, T 为采样周期。选择加到电机驱动模块上的 PWM 波的占空比 U 作为系统输出。设 R 为一常值, $R = 180$, 通过计算 e/R 和 ec/R 将系统误差及其变化归一化, 并将其在闭区间 $[0, 1]$ 内分成若干等级以完成 Fuzzy 量化。本章的算法中 Fuzzy 控制系统选取如下形式的误差 E 及误差变化 EC 的归一 Fuzzy 量化:

$$E = \begin{cases} 5\text{sgn}(e) & \text{if } |e/R| \geq 0.4 \\ 4\text{sgn}(e) & \text{if } |e/R| \geq 0.2 \\ 3\text{sgn}(e) & \text{if } |e/R| \geq 0.1 \\ 2\text{sgn}(e) & \text{if } |e/R| \geq 0.05 \\ 1\text{sgn}(e) & \text{if } |e/R| \geq 0.02 \\ 0 & \text{if } |e/R| < 0.02 \end{cases} \quad (12-3)$$

$$EC = \begin{cases} 5\text{sgn}(ec) & \text{if } |ec/R| \geq 0.4 \\ 4\text{sgn}(ec) & \text{if } |ec/R| \geq 0.2 \\ 3\text{sgn}(ec) & \text{if } |ec/R| \geq 0.1 \\ 2\text{sgn}(ec) & \text{if } |ec/R| \geq 0.05 \\ 1\text{sgn}(ec) & \text{if } |ec/R| \geq 0.02 \\ 0 & \text{if } |ec/R| < 0.02 \end{cases} \quad (12-4)$$

其中, E 及 EC 分别为控制系统误差及其变化的 Fuzzy 论域; $\text{sgn}(x)$ 为符号函数, $x > 0$ 时, $\text{sgn}(x) = 1$, $x < 0$ 时, $\text{sgn}(x) = -1$ 。Fuzzy 论域的划分应满足下列条件: ① 无混叠现象, 以免控制规则不惟一; ② 无空挡现象, 以防止出现失控点; ③ 零挡划分应反映控制精度。

2. 带修正函数的 Fuzzy 规则自调整

设 Fuzzy 控制器的输出为 U , 则

$$U = \begin{cases} U_0 + \alpha E & |E| > E_m \\ U_0 + \alpha E + (1 - \alpha)EC & |E| \leq E_m \end{cases} \quad (12-5)$$

式中, α 为修正因子; E_m 为阈值。本系统中取 $E_m = 30$ 、 $U_0 = 128$ 为系统稳态时的输出, 误差大时, 控制作用主要应表现为尽快消除误差, 提高响应速度, 故需对误差给予较大的权重; 当误差小时, 为减小系统响应的超调使之尽快进入稳态, 需对误差变化给予较大权重。基于这种控制策略, 建立如下修正函数:

$$\alpha = K \left| \frac{e}{R} \right|^P \quad (12-6)$$

式中, 当 $P < 1$ 时, 修正因子 α 随误差 e 增大递增较快, 控制规则中 E 的作用相对加强; 当 $P > 1$ 时, 情况相反, 控制规则中 EC 的作用加强, 且 P 值越大, 其作用越强。通常取 $0.5 \leq P \leq 3$ 。参数 K 的取值与参数 P 的取值有关, 通常为了使误差初值较大时过程响应较快, 取 $K > 1$ 。但还需考虑 $|E| = E_m$ 时应满足 $(1 - \alpha) \geq 0$ 的要求, 可以从上式求得 K 的取值范围为: $1 \leq K \leq |R/e_m|^P$ 。

通过修正因子的变化, 很容易实现不同状态下过程响应对误差 E 及其变化 EC 的分离加权要求, 达到灵活调整模糊控制规则的目的。

3. 模糊控制算法的实验

通过对修正因子中的参数 K 、 P 以及比例因子 K_u 等参数进行单纯形寻优得到一组优化参数: $K = 50$ 、 $P = 0.8$ 、 $K_u = 0.5$, 将这组参数输入到模糊控制器中, 在精度和可靠性方面都达到了预期的控制效果。

12.4 视觉系统

通过悬挂在平台上部的 CCD 摄像机采集机器人作业环境的图像, 并依此计算双臂机器人的姿态和被捕获目标的位置和姿态, 为机器人姿态调整及捕获过程提供定位信息。以往的系统中大都采用黑白 CCD 及图像处理卡, 输入的是灰度图像。机器人的姿态和目标姿态大都采用对各关节以特征位置打标记, 并求出各标记中心坐标, 从而按各关节约束来求得机器人姿态和目标位置、姿态。

12.4.1 FFSR 地面实验平台全局视觉方案

视觉系统由 CCD 摄像机、图像采集卡和图像处理计算机组成。CCD 摄像机安放在平台的正上方, 其输出送视觉处理计算机内部的图像采集卡, 而计算机控制采集卡的工作, 并完成识别定位计算, 其计算结果通过局域网送给运动规划子系统。图 12-5 为视觉系统硬件布置实物照片。运动规划子系统接受视觉处理计算机提供的有关机器人各关节的坐标, 计算出机器人当前的姿态, 并根据所得姿态和捕获目标位置、姿态产生相应的策略, 并通过通信模块向机器人发送姿态变换命令。其中机器人为 6 自由度双臂机器人, 如图 12-6 所示。在无变形的情况下每一臂要得到其姿态至少需要知道臂上两个关节的坐标, 加上本体中心坐标 7 个点的坐标, 在本系统中选择如图中黑点所示 7 个关节进行标记, 而臂上的其余的两个关节由 $L1$ 、 $L2$ 、 $R1$ 、 $R2$ 分别计算出。所选中的每一个点贴一种颜色块作

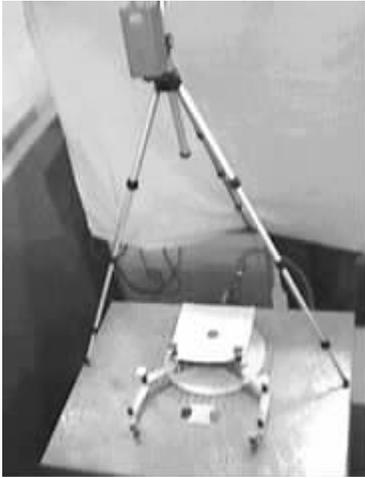


图 12-5 视觉系统硬件布置实物照片

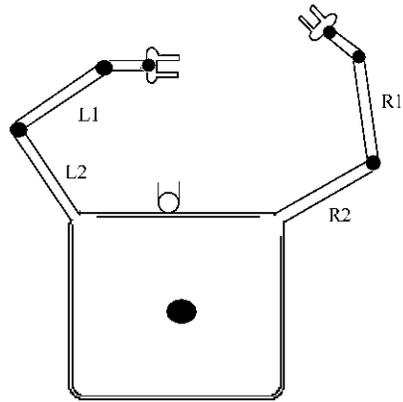


图 12-6 彩色标志布置图

标记。

1. FFSR 实验平台全局视觉硬件系统设计

FFSR 全局视觉系统硬件系统由视觉处理计算机,图像采集卡及彩色 CCD 摄像机组成。CCD 摄像机悬挂在实验平台正上方,其视野覆盖整个平台。CCD 的输出作为安装在计算机内的图像处理卡的输入。图像采集卡采用加拿大 MATROX 的 METEOR II 彩色实时采集卡,每秒可采集 60 帧 16 位 RGB 图像。视觉处理计算机为 IBM 兼容微型计算机。视觉处理计算机上运行全局视觉软件系统,控制图像卡工作,进行识别与定位得出各个目标点的位置信息,并把结果通过局域网送给系统内其他计算机。

2. FFSR 实验平台全局视觉软件系统

FFSR 地面实验平台中,全局视觉为整个系统提供环境信息,包括机器人本体的姿态,目标障碍物的位置、形状等信息。全局视觉由上向下俯视整个实验平台,并自主跟踪浮游的空间机器人及目标,保证整个环境在视野范围内。根据 FFSR 的具体要求,其完成的功能有以下几个方面。

(1) 机器人本体识别、定位及姿态分析。机器人的姿态可以通过机械臂各关节的位置和相邻连杆之间的夹角来确定。机器人本体的位置与姿态通过其在平台坐标系中的方位来确定和表示。

(2) 目标和障碍物的识别、定位,并对所有目标具有学习能力。

(3) 序列图像的分析。根据目标及机器人的运动,对运动目标的位置进行预测,从而缩小特征点的搜索范围,减少帧处理时间,并提高系统的准确性。

此外,全局视觉还应有高的实时性和稳定性,因为在地面实验平台中研究重点在于机器人浮游状态下的动力学特征,因此,对系统进行简化,暂不考虑障碍物的回臂问题。

视觉系统的任务就是求出各标记块的中心,并能实时地跟踪。在系统中每一个标记块颜色是固定的单一的,因此,对标记块的跟踪定位问题就是在每一帧输入图像中分离出已知颜色的区域,并求它们各自的中心。为此利用局部颜色直方图方法来实现跟踪及定位。

1) 局部颜色直方图聚类法

自然界中的颜色可以用三基色(红,绿,蓝)表示。计算机中颜色的表示也是一样,不过可选择的基色组合有多种方案,如RGB、YUV等。其中RGB是最常见的。绝大多数图像采集卡都支持该模式,在本系统中也采用这种模式。

所谓局部颜色直方图方法是指,生成局部区域上图像直方图,并通过比较模版(参考图像)与目标图像之间局部直方图值的相似性来检测图像中物体位置的方法。具体实现上借用 Swain 等人提出的颜色直方图度量。

首先在参考图像上要探索的物体周围局部区域上选择包含物体的正方形小区间,并在该区间上计算颜色直方图作为参考图像直方图。直方图按如下计算:首先把 RGB 空间的各轴分成 Q 个等间隔区间,计算三维直方图。用上述方法计算所得直方图为 $M_j (j = 1, 2, \dots, I), I = Q^3$ 。用同样的方法在目标图像上局部区域上计算目标直方图 H_j , 则参考图与目标被选区域相似度选用:

$$S_{HM} = \sum_{j=1}^I \min(H_j, M_j) \quad (12-7)$$

作为度量。在 FFSR 实验平台系统中每一个待跟踪目标都具有单一的颜色,因此,可以选择更为简单的相似性度量。下面做具体介绍。

2) S_{HM} 的定义及目标定位方法

在 FFSR 实验系统中每个关节上的标记都选用单一颜色,因此, S_{HM} 的计算可以进一步简化。

设系统所使用的不同颜色个数为 N , 那么、系统总共需辨别 $N+1$ 种不同颜色。因此,只要选 $I = N+1$ 即可。每一种颜色分别对应一组 RGB 区间 $(\Delta r_i, \Delta g_i, \Delta b_i)$, 这相当于 RGB 空间中的长、宽、高分别为 $\Delta r_i, \Delta g_i, \Delta b_i$ 的“箱子”。而计算相应直方图就是要计算落在该“箱”内的像素个数。而计算标记块位置就是要计算该标记颜色区域的中心坐标。为便于计算颜色块中心,给每一个“箱”一个惟一的标记(选用一个整数)。其实质是构造如下函数: $U(R, G, B) \rightarrow (0, L)$

$$U(\Delta r_i, \Delta g_i, \Delta b_i) = L_i \quad L_i \in [0, L] \quad (12-8)$$

式中, $L+1$ 为颜色标记个数, 0 对应背景。

直方图的计算和像素的标记过程是同时进行的:对图像进行扫描,获得每一个像素的 RGB 值,判断该像素落在哪一只“箱子”,并给予相应的标记。最后计算每一只“箱子”内像素个数所得结果就是该箱子对应的直方图值。因此,设图像位置 (x, y) 处像素的颜色为 $COL(x, y)$, 若

$$COL(x, y) = (r, g, b) \quad (12-9)$$

则第 l 个标记色参考直方图 M_l 为:

$$M_l = | \{ (x, y) \mid U(r, g, b) = L_l \} | \quad (12-10)$$

同样方法可计算目标图像上第 l 个标记色直方图 H_l 。因此,与第 l 个标记色相关的相似度 $S(H_l, M_l)$ 为:

$$S(H_l, M_l) = \min(H_l, M_l) \quad (12-11)$$

经过标记后的图像画面已被有限个整数标记分割。此时,用式(12-8)可计算出每一个具有相同标记的连通区域的中心坐标 $[CX_k, CY_k]$ 。

$$CX_k = \sum_{y=1}^N \sum_{x=1}^M xf(x,y)/A \quad CY_k = \sum_{y=1}^N \sum_{x=1}^M yf(x,y)/A \quad (12-12)$$

式中, N, M 为图像纵、横像素个数; $A = MN$; (x, y) 为像素坐标; $f(x, y)$ 是判别像素 (x, y) 是否落在具有 l 标记的第 k 个连通区域 (k, l) 的特征函数:

$$f(x, y) = \begin{cases} 1 & \text{if } (x, y) \in D(k, l) \\ 0 & \text{if } (x, y) \notin D(k, l) \end{cases} \quad (12-13)$$

12.4.2 视觉系统软件的工作原理

视觉系统工作大体上分两个模块:训练过程与识别过程。

1. 训练过程

由于系统工作环境的光照在不同时间可能不一样,因此每次运行时系统首先要对各个标记的颜色进行认定工作,其实质就是在当时的光照条件下生成各关节的参考直方图 M 和构造函数 U 的具体值。方法如下:在指定的图像目标周围固定大小区域(大于关节标记尺寸)内所有像素的RGB值,并计算出该区域内R、G、B各自的最小最大值集合:

$$SET_l = [(R_{\min}, R_{\max}), (G_{\min}, G_{\max}), (B_{\min}, B_{\max})] \quad (12-14)$$

由 SET_l 很易构造第 l 个关节所对应的“箱子” $(\Delta r_l, \Delta g_l, \Delta b_l)$,只要选:

$$\Delta r_l \equiv [R_{\min}, R_{\max}] \quad \Delta g_l \equiv [G_{\min}, G_{\max}] \quad \Delta b_l \equiv [B_{\min}, B_{\max}]$$

即可。同时构造函数 U :

$$U(r, g, b) = L_l \quad (r \in \Delta r_l \wedge b \in \Delta b_l \wedge g \in \Delta g_l) \quad (12-15)$$

并把这些设定值和落在该“箱”内的像素个数 M_l 作为第 l 个标记的参考直方图进行保存供识别时使用。

2. 识别过程

识别过程是视觉系统的核心,在这里利用训练过程中产生的各个 M_l ,在实时采集的图像上识别和定位各个关节上的标记。识别定位分两步进行:图像标记过程和定位过程。

图像标记过程中根据设置过程中生成的函数 U 对各个像素 (x, y) 按下述方法进行标记。设 $U(r, g, b)$ 为像素 (x, y) 的颜色, r, g, b 分别为红绿蓝三基色分量,则像素 (x, y) 的标记 $L(x, y)$ 由下式确定:

$$L(x, y) = U(r, g, b) \quad (12-16)$$

对图像区域内所有的像素按式(12-16)进行标记后图像区域变成由背景及其他几个目标标记分割的图像,称之为标记图像。由于成像设备本身的原因或因外界因素的影响(如光照不均等),标记图像上每一种合法标记的连通区域不是惟一的,因此用式(12-7)的相似性判断来识别各个目标所对应的标记区。过程如下:

(1) 选择一个合法标记 L_l ;

- (2) 找到具有标记 L_l 的一个连通区域 $D_l(I)$;
- (3) 计算 $D_l(I)$ 内像素个数 $H_l(I)$, 计算 $D_l(I)$ 与 M_l 的相似度 $S(H_l(I), M_l)$;
- (4) 若图像上还有未计算 S 值的具有标记 L_l 的区域返回(2), 否则执行(5);
- (5) 求 $\{S(H_l(I), M_l)\}$ 的最大值 $S_{\max}(I)$, 若 $S_{\max}(I) \geq TH$, 则 $S(H_l, M_l) = S_{\max}(I)$, 否则 $S(H_l, M_l) = 0$;
- (6) 如果还有未考察的标记色, 则返回(1), 否则结束;

与 $S_{\max}(I)$ 所对应的 $D_l(I)$ 即为与 l 标记色对应的目标, 在图像上识别出各个目标以后可以用式(12-8) 计算各目标区域的中心坐标。其中 TH 为相似度判别的阈值, 可以选择 M_l 的简单函数如: $\lambda M_l (\lambda \in (0, 1])$ 。

3. 系统运行环境及实验结果

本系统图像采集卡每秒采集 60 帧图像, 视觉处理程序每秒处理(30 ~ 47) 个周期(一个周期包括: 一帧图像的识别定位处理和结果送系统内其他计算机) 经 100 次运行, 平均周期为 32 帧 /s。100 帧的平均识别为(各个关节识别正确)96.4%。定位最大误差(在正确识别情况下)0.9cm, 平均 0.3cm。

12.5 操作台

操作台采用一台 IBM-PC 机。它通过视觉反馈系统提供的环境信息形成三维图像, 为操作员提供操作平台。操作员可以通过键盘、鼠标等外部设备输入控制命令, 操作系统把命令传给规划器, 由规划器去执行。

大量 FFSR 的实验研究表明, 由于空间作业环境的复杂多变性, 让空间机器人完全自主地完成各种空间作业任务是不现实的, 它往往需要航天员或地面操作人员进行必要的监控和遥操作。同时, 由于受天地之间的通信时延和微重力影响, 用航天员的肉眼作为操作依据往往容易造成误判, 致使机器人操作的失败或设备部件的损坏。因此, 需要采用高层人工监控协调控制与底层局部自主行为控制相结合的共享控制技术是解决上述问题得一种有效方法。

操作台的主要功能是: ① 操作台提供一个可视化的 FFSR 地面平台控制台的现场, 视觉系统将采集到的视觉信息通过网络服务器实时传送到控制台上; ② 地面指挥人员根据操作台显示的现场信息驱动机器人完成任务。

12.5.1 操作台显示信息

操作台是将机器人现场上的视觉信息展现给地面指挥系统。控制台适用仿真软件 OpenGL 实时显示现场信息的。

(1) OpenGL 概念: 所谓 OpenGL 是指在 SGI, Microsoft 等多家世界著名计算机公司倡导下, 基于 SGI 的 GL 标准, 制定一个独立于操作系统、通用共享的和开放式的三维图形标准。目前, OpenGL 已被公认为是高性能图形和交互式视景处理的标准, 广泛应用于可视化、动化及虚拟显示等技术领域。OpenGL 图形库是一个硬件图形发生器的软件接口, 主要有基本库、实用库(GLU) 和辅助库(GLAUX) 组成。基本库包括 100 多个图形操

作函数,其中核心函数有 115 个,其前缀为 gl;GLU 共有 43 个函数,其前缀为 glu,是更高级功能的函数,如绘制复杂曲面、多边形分割和坐标变换等;GLAUX 共有 31 个函数,前缀为 aux,使一些特殊函数,如窗口管理、时间处理和绘制复杂三维物体等。另外设有专用前缀的 Win32API 函数。

(2) OpenGL 建模:OpenGL 能构造和渲染物体三维模型,生成动态、逼真和可交互的真实感图形,并进行三维图形交互软件的开发。其主要基本功能有模型绘制、坐标变换、颜色模式设置、光照和材质设置、图像功能、实时动画等功能。

12.5.2 操作台命令

操作台可以向服务器发送下列的命令来控制机器人:

(Move {Joint_n, Angle}):把命令机器人的指定关节转动 Angle 角度转送到操作台。

其中 Joint_n 为机器人的 n 关节,Angle 表示关节所转的角度和方向。

操作台所发射的命令有:

(1) 关节驱动命令:在机器人本体及机械臂姿态出现偏差时驱动某一具体关节转动某一角度来调节机器人本体姿态和机械臂姿态。

(2) 捕捉目标命令:选择合适的捕捉方式连续发送关节驱动命令来驱动各个关节去捕捉目标。

操作台界面如 12-7 图所示。其中典型的命令有:

- (1) 系统初始化。
- (2) 设零点。其功能是统一机器人和操作台中的零点。
- (3) 电磁铁吸合与断开。
- (4) 显示目标命令。提取目标信息,将目标显示在规划器中。
- (5) 双臂外展 20° 。机器人双臂的大臂关节,肘关节,手关节各外转 20° 。

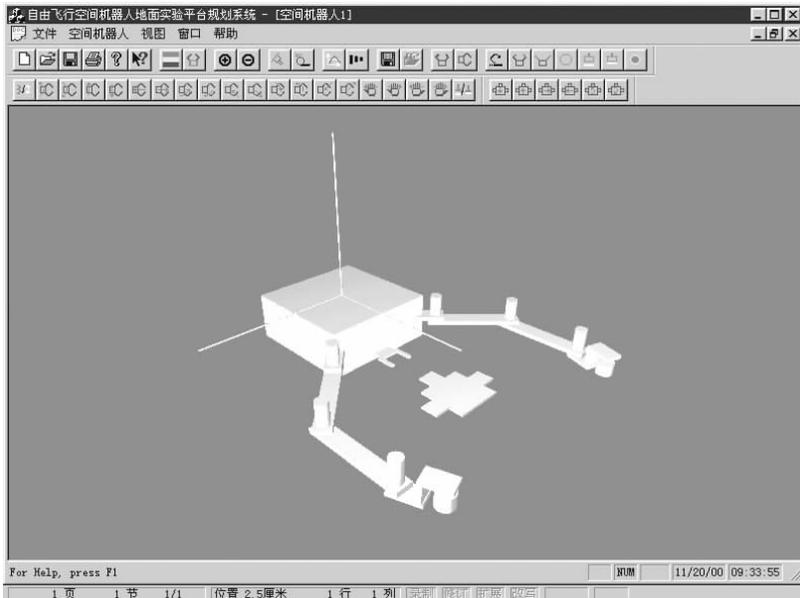


图 12-7 操作台操作界面

- (6) 双臂内收 20° 。机器人双臂的大臂关节,肘关节,手关节各内转 20° 。
- (7) 机器人左大臂外转 5° 。
- (8) 机器人左臂肘外转 5° 。
- (9) 机器人左手外转 5° 。
- (10) 机器人右臂外转 5° 。
- (11) 机器人右臂肘外转 5° 。
- (12) 机器人右手外转 5° 。
- (13) 参数辨识命令。利用神经网络训练的权值计算机器人本体质量。
- (14) 捕捉目标命令。驱动机器人右臂捕捉目标,分为无姿态控制捕捉目标方式和有姿态控制捕捉目标方式。
- (15) 操作目标命令。驱动机器人左臂操作目标,分为无姿态控制操作目标方式有姿态控制操作目标方式。
- (16) 对接目标命令。驱动机器人左臂操作目标,分为无姿态控制对接目标方式有姿态控制对接目标方式。

12.6 规划系统

规划系统采用一台 IBM-PC 机。它的主要任务是通过视觉反馈系统提供的环境信息和接收操作系统发送的命令完成任务规划和路径规划。并通过无线发射模块向机器人发出执行命令,以完成具体系列动作。

规划器的主要功能是通过视觉系统传来的信息自主地完成对机器人的各种行为的监视和控制,也可以接收操作台发送来的控制命令执行操作员对机器人的遥控操作。规划器应具有相当的智能和机器人在线判断功能,能够按要求自主地操作。

规划器包含逻辑和判断功能。它担负着协调和执行用户的命令。事实上,规划器完成下列功能:协调本体和机械臂的运动、选择捕捉方向、计算轨迹、不同控制方式之间的转换和监视整个系统状况等。它既可以接受从操作台传来的命令,也可以接受从视觉系统传来的信息。当接受一个新的命令时,规划器将负责选择一系列合适的动作。为了实现这一功能,它包含一些控制算法和逻辑准则。一旦决定了一系列动作,规划器将执行它们,通过调用程序来计算合理轨迹和配合必要的不同控制形式之间的转换来实现。

下面介绍该系统的规划器所采用的一些算法和逻辑准则,规划器应用它们来进行判断和规划操作。

12.6.1 捕捉策略

对于 DFFSR 捕捉目标的策略,该系统采用双臂先后协调捕捉的策略,捕捉分为两个过程完成:第一个过程是用右臂捕捉目标的右手柄,第二个过程是用左臂捕捉目标的左手柄。这种捕捉策略将双臂同时捕捉改成双臂先后捕捉,这种捕捉方法有效地减少捕捉瞬间机器人和目标之间的相互作用力和动量交换,保证机器人可以安全、可靠地捕捉目标。在这里,第一过程与第二过程可以相互交换,但捕捉规划算法还是一样的。在机器人用右臂捕捉目标上的右手柄之后,右臂的最后一个连杆和目标可看做为一体,形成一个新的连

杆,然后再用左臂捕捉目标的左手柄。

12.6.2 基于广义雅可比矩阵捕捉目标的运动规划

地面机器人的雅可比矩阵表示了惯性坐标系下机器人末端效应器的速度与关节角速度的关系,从而导出分解运动速度控制。但是对于自由飞行空间机器人,由于其基座不固定,机械臂的运动会对本体的位置和姿态有影响,从而也改变了机械臂的位置和姿态,故而不能直接应用雅可比矩阵。应采用在空间微重力条件下反映机器人末端效应器的速度与关节角速度关系的广义雅可比矩阵。广义雅可比矩阵与地面固定基座机器人的雅可比矩阵不同,雅可比矩阵仅与机器人的几何参数有关,而广义雅可比矩阵不仅与机器人的几何参数有关,还与机器人的惯性参数,如质量、转动惯量等有关。因此有:

$$\dot{\mathbf{P}}^k = \mathbf{J}^{*k} \dot{\boldsymbol{\theta}}_m^k \quad (12-17)$$

式中, \mathbf{J}^{*k} 为 FFSR 第 k 个机械臂末端运动速度的广义雅可比矩阵; $\dot{\mathbf{P}}^k$ 为机器人第 k 个机械臂末端效应器的速度; $\dot{\boldsymbol{\theta}}_m^k$ 为机器人第 k 个机械臂各关节速度。 \mathbf{J}^{*k} 的具体形式为:

$$\mathbf{J}^{*k} = \mathbf{J}_m^k - \mathbf{J}_S^k \mathbf{I}_S^{-1} \mathbf{I}_m \quad (12-18)$$

式中, \mathbf{I}_S 和 \mathbf{I}_m 分别为本体和机械臂的广义惯性矩阵, \mathbf{J}_S^k 和 \mathbf{J}_m^k 分别为与本体和机械臂的运动有关的广义雅可比矩阵。利用 FFSR 的广义雅可比矩阵进行分解运动速度控制,可以准确地捕捉目标。

12.6.3 基于姿态受限广义雅可比矩阵捕捉目标的运动规划

为了保证 FFSR 的通信系统和电源系统的正常工作,必须确保机器人搭载的碟形天线和太阳能帆板的定向,因此姿态稳定对于 FFSR 而言是至关重要的,即应该确保机器人本体的姿态在机械臂运动过程中保持不变。为此,对于本系统的双臂冗余 FFSR,采用多臂 FFSR 末端效应器任务空间的姿态受限广义雅可比矩阵实现捕捉目标的协调运动规划,使 FFSR 完成空间作业的同时,通过双臂协调运动,相互抵消对本体的姿态干扰。

将 FFSR 的关节空间看为由各个机械臂的关节角组成。当第 k 个机械臂完成给定任务时,利用其他机械臂进行协调运动,使得机器人工作时不对本体产生姿态干扰。因此有:

$$\dot{\mathbf{P}}^k = \mathbf{J}_{AR}^k \dot{\boldsymbol{\theta}}_m^k \quad (12-19)$$

式中, \mathbf{J}_{AR}^k 为 FFSR 第 k 个机械臂的基于多臂协调的姿态受限广义雅可比矩阵,其具体形式为:

$$\mathbf{J}_{AR}^k = \mathbf{J}_m^{k,k} - \mathbf{J}_m^{k,l-k} [\mathbf{I}_m^{l-k}]^+ \mathbf{I}_m^k \quad (12-20)$$

式中, $\mathbf{J}_m^{k,k}$ 和 $\mathbf{J}_m^{k,l-k}$ 分别为矩阵 \mathbf{J}_m^k 和 \mathbf{J}_m^k 中第 k 个机械臂各关节角对应的部分形成的子矩阵; \mathbf{I}_m^{l-k} 和 $\mathbf{J}_m^{k,l-k}$ 分别为矩阵 \mathbf{I}_m 和 \mathbf{J}_m^k 中去掉第 k 个机械臂各关节角对应的部分形成的子矩阵; $[\cdot]^+$ 为 $[\cdot]$ 的伪逆矩阵。利用 FFSR 第 k 个机械臂的基于多臂协调的姿态受限广义雅可比矩阵进行分解运动速度控制,在捕捉目标的同时可以控制本体的姿态不受干扰。

12.7 无线通信系统

通信系统的主要任务是把规划器产生的策略和动作命令通过通信系统向机器人发送。本系统采用单向通信方式,利用商用的 R/F 模块。通信模块为 BIM-418F(如图 12-8 所示)。发射模块接在主机(规划器)的串行口上,接收模块接在机器人模型的 CUP 板上。其工作原理如图 12-9 所示。

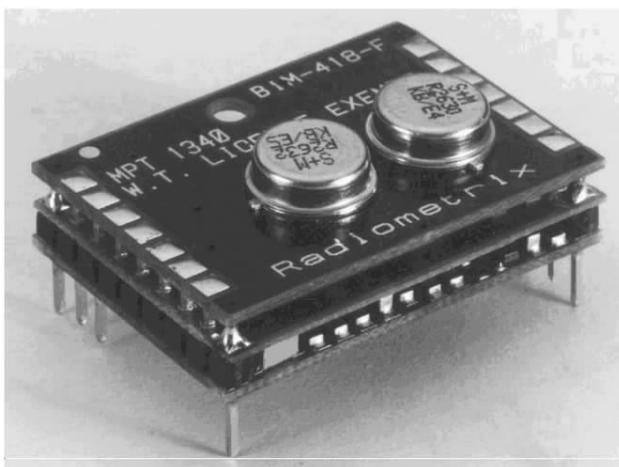


图 12-8 BIM-418F 通信模块的实物照片

本系统中采用商用无线通信模块 BIM-418F 作为规划器和机器人模型之间的通信设备。BIM-418F 使用简单,既可用于发射也可用于接收。规划器和机器人之间的传输数据主要是各电机的转角、方向和速度以及电磁铁的开关命令。为了提高对电机的控制精度,采用 16 个字节(即 8 个字)的命令字格式。具体解释如下:

第 0 和 4 个字分别为两个电磁铁的命令字,第 1、2、3、5、6、7 分别为 6 个关节电机的命令字。每个字的具体格式如下:

D ₁₅	D ₁₄	D ₁₃	D ₁₂	D ₁₁	D ₁₀	D ₉	D ₈	D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
NA	电机的速度			命令		给定的电机转角									

其中:NA 表示此位不用;

位 D₁₄D₁₃D₁₂ 为电机的转速,共有 7 级;

位 D₁₁D₁₀ 为发给电机的命令:

D₁₁D₁₀ = 01:表示顺时针旋转(对电机)或吸合(对电磁铁);

D₁₁D₁₀ = 10:表示逆时针旋转(对电机)或断开(对电磁铁);

D₁₁D₁₀ = 00:表示停止;

D₁₁D₁₀ = 11:表示使当前的位置为电机的零位置;

位 D₉ ~ D₀ 为电机转角。

上述的通信接收模块安装在机器人的本体上,与单片机 80C196KC 的串行输入口相连。通信发射模块接在规划器的串行口,规划器根据机器人和目标的初始位置,决定机器人各个关节的运动,并且把相应的命令按上述的命令字格式通过发射模块发射给机器人,

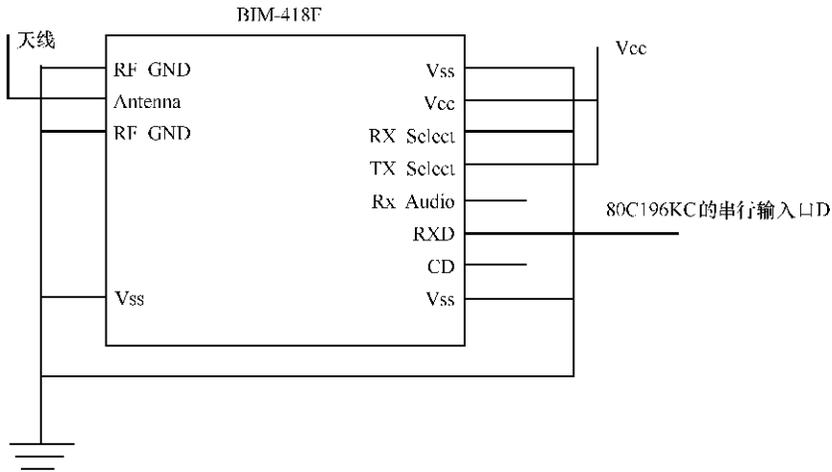


图 12-9 通信接收模块原理图

机器人上的接收模块接到命令后,经过译码,驱动相应关节的电机转动以完成给定的任务。

12.8 网络系统

12.8.1 网络系统构成

考虑到整个系统的易用性和客户端开发平台无关性,整个的 FFSR 地面平台的网络系统采用基于网络通信的客户-服务器模式。如图 12-10 所示,表示了 FFSR 地面平台的网络系统各个组成部分之间的关系。整个系统由两部分组成,第一部分为空间机器人地面平台的网络服务器。它负责控制整个任务的进行,包括向客户端发送机器人及目标的视觉信息,操作台的命令信息等功能。第二部分是客户端。客户端由操作台和规划器组成。因为整个 FFSR 地面平台系统是基于网络连接的,所以服务器和客户程序可以分别运行在因特网上的多台计算机上。因此,为了能让地面指挥系统及时地了解现场的情况,设计了用于显示现场情况的操作台。而且,对于客户端程序的开发,并不一定要限制在统一的平台下。客户端程序不仅可以在 Windows 下,也可以基于 Linux、Unix 等操作系统进行开发,只要

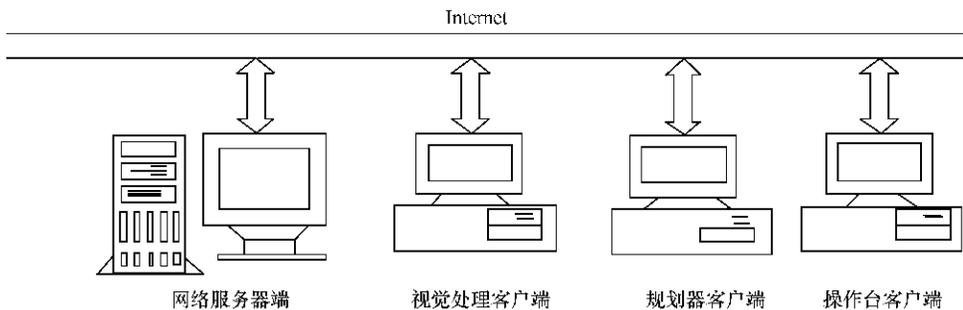


图 12-10 网络系统结构图

遵循基本的 Socket 网络协议,就可以顺利进行机器人控制。由于规划器程序主要负责向机器人发送命令,这里就不做过多的介绍,把重点放在服务器和操作台的开发上。

12.8.2 服务器端的框架

FFSR 地面平台的网络服务器为操作台和规划器提供了一个完善的信息中转环境,操作台和控制台通过独立的客户端程序利用计算机网络分别和服务器连接。整个机器人系统服务器主要由两个模块组成:网络通信和信息处理模块。主要处理所有客户端和服务器的采用 ASCII 字符串形式的网络通信。每个客户程序通过 TCP/IP 协议和服务器连接,客户端程序通过向服务器发送控制命令字,来控制机器人的运动。

由于整个 FFSR 地面平台的网络系统采用客户-服务器模式,客户和服务器之间通过计算机网络传递机器人图像信息和控制台对机器人的控制命令。客户端程序和服务器程序可以分散在同一局域网内的不同计算机上,彼此之间都是相互独立的。服务器的运行采用中断等待的方式。只要有任意客户端向服务器发送控制命令,服务器便接收并处理控制命令。

在机器人开始动作之前,每一个客户都必须和服务器建立一个网络连接。具体的初始化过程按照下面的规范进行。

(1) 客户程序发送如下的信息到指定的服务器端口地址:

(init Port_name)

(2) 如果服务器没能正确地分配一个给客户端,那么,服务器向客户端发送如下消息:

(int Connection_Failed),如果收到这个消息,客户端程序需要检查一下和服务器连接时的情况,可以再次申请和服务器的连接。

客户端程序可以向服务器发送下列的命令来控制机器人:

(Move {Joint_n, Angle}):把命令机器人的指定关节转动 Angle 角度转送到空操作台。

12.8.3 控制台结构

为了提供一个可视化的机器人现场环境,FFSR 地面平台控制台可以通过和服务器的网络连接,实时地接收服务器传送来的现场机器人的视觉信息。监视器提供了地面平台环境,可以看到机器人和目标在现场的运动。它主要由以下几个部分组成。

1. 和服务器的初始化连接

控制台也作为一个客户程序和服务器连接,连接采用下面的规则。

(1) 控制台向服务器端口发送如下的初始化连接信息:

(init initdisplay),表示向服务器申请连接。

(2) 当服务器成功为控制台分配了内存,且获取了控制台的端口地址,服务器将向控制台的端口地址返回如下信息:

(init Connection_Success),表示服务器已经成功和控制台建立网络连接。

(3) 如果服务器没有成功为当前申请连接的控制台分配内存,服务器将向控制台返回信息:

(init Connction_Failed),表示服务器和控制台之间的连接请求失败。如果收到这个消息,控制台可以再次申请和服务器连接。

2. 操作台和机器人之间的通信

操作台的主要功能是机器人现场的视觉信息向机器人发送控制命令。客户端程序可以向服务器发送下列的命令来控制机器人:

(Move {Joint_n,Angle}):把命令机器人的指定关节转动 Angle 角度转送到操作台。

12.9 气浮系统

气浮系统由气浮平台、气室和空气压缩机组成。在空气压缩机供气的条件下,可使机器人模型在气浮平台上浮起 0.15mm,此时空气压力为 3×10^5 Pa,流量为 150L/min。气浮平台表面面积为 $750\text{mm} \times 600\text{mm}$,平面度为 0.05mm,表面粗糙度为 $Ra = 1.6$,材料为 ZL303。气浮平台下安装支架,支架设置水平调节机构,支架尺寸为 $750\text{mm} \times 600\text{mm} \times 750\text{mm}$ 。空气压缩机的工作压力为 8×10^5 Pa,流量为 192L/min。其实物照片如图 12-11 所示。



图 12-11 气浮系统实物照片

12.10 仿真实验

12.10.1 捕捉目标

采用右臂捕捉目标,分两种情况:一是无姿态控制的情况,采用广义雅可比矩阵在机械手运动对本体运动有干扰的情况下,准确地捕捉目标,如图 12-12 所示;另一种情况是有姿态控制,即采用姿态受限广义雅可比矩阵,在机械手准确捕捉目标的同时,保证机器人本体的姿态不变,如图 12-13 所示。



图 12-12 无姿态控制捕捉目标

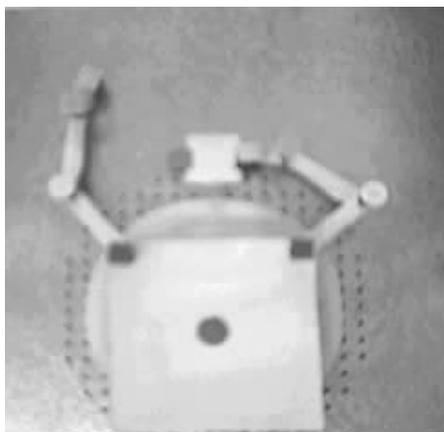


图 12-13 有姿态控制捕捉目标

12.10.2 操作目标

当右臂捕捉目标之后,采用左手对目标进行操作,此时因为右手和目标连在一起,要重新计算雅可比矩阵,也分两种情况:一是无姿态控制的情况,采用广义雅可比矩阵在机械手运动对本体运动有干扰的情况下,准确地操作目标,如图 12-14 所示;另一种情况是有姿态控制,即采用姿态受限广义雅可比矩阵,在机械手准确操作目标的同时,保证机器人本体的姿态不变,如图 12-15 所示。

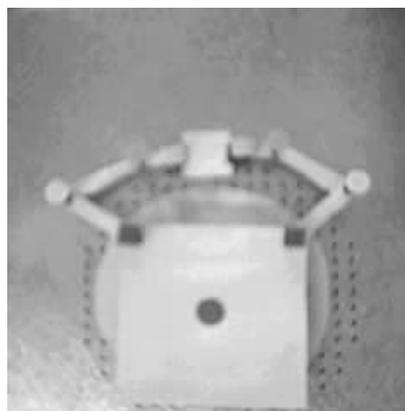


图 12-14 无姿态控制操作目标



图 12-15 有姿态控制操作目标

12.10.3 对接目标

当右臂捕捉目标之后,用右手夹持目标往固定臂上对接,此时因为右手和目标连在一起,要重新计算雅可比矩阵,也分两种情况:一是无姿态控制的情况,采用广义雅可比矩阵在机械手运动对本体运动有干扰的情况下,准确地对接目标,如图 12-16 所示;另一种情况是有姿态控制,即采用姿态受限广义雅可比矩阵,在机械手准确对接目标的同时,保证机器人本体的姿态不变,如图 12-17 所示。

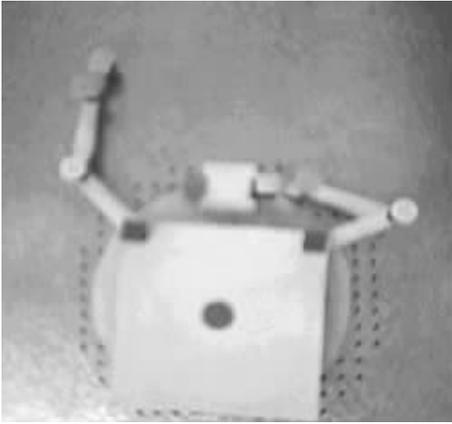


图 12-16 无姿态控制对接目标

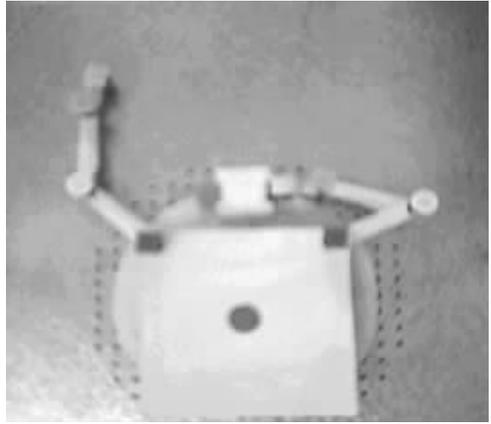


图 12-17 有姿态控制对接目标

12.11 小 结

本章介绍了哈尔滨工业大学研制的双臂 FFSR 地面试验平台。它由机器人模型、视觉系统、无线通信系统、网络系统、规划系统、气浮系统和操作台 7 个模块组成。并分别对各个模块进行了具体阐述。实验证明该平台可以很好地模拟空间微重力环境，并可以验证 FFSR 在空间微重力环境下的运动学和动力学特性。

参 考 文 献

- 1 Bejczy AK, Venkataraman S T. Introduction to the Special Issue on Space Robotics. *IEEE Trans. Robotics Automat.* 1993, 9(5):521 ~ 523
- 2 Nenchev D, Umetani Y. Analysis of a Redundant Free-Flying Spacecraft /Manipulator System. *IEEE Trans. Robotics Automat.* 1992,8(1):1 ~ 6
- 3 Nenchev DN. A Controller for a Redundant Free-Flying Space Robot with Spacecraft Attitude/Manipulator Motion Coordination. *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 1993:2108 ~ 2114
- 4 Papadopoulos E. On the Design of Zero Reaction Manipulators. *Transactions of the ASME Journal of Mechanical Design.* 1996,118:372 ~ 376
- 5 Papadopoulos E, Moosavian SA. Dynamics and Control of Space Free-Flyers with Multiple Manipulators, *Advanced Robotics.* 1995,9(6):603 ~ 624
- 6 Papadopoulos E, Dubowsky S. On the Nature of Control Algorithms for Free-Floating Space Manipulators, *IEEE Trans, Robotics Automat.* 1991,7(5):750 ~ 758
- 7 Papadopoulos E, Dubowsky S. Coordinated manipulator/spacecraft motion control for space robotic systems. *IEEE Proc. Robotics Automat.*, April 1991:1696 ~ 1701
- 8 Wilson E, Rock SM. Neural-network control of a free-flying space robot. *Simulation.* 1995,65(2):103 ~ 115
- 9 Weigl HJ, Alexander HL. Robot free-flyers in space extravehicular activity, *Proceedings of SPIE'93*, 1993, 458 ~ 469
- 10 Russakow J. Experiments in Manipulation and Assembly by Two-arm Free-flying Space Robot. PhD thesis, Stanford University, Stanford, December 1995
- 11 Machida K, Toda Y. Maneuvering and Manipulation of Flying Space Telerobotics System. *Proc. IEEE/RSJ on Intelligent Robotics and systems.* 1992:3 ~ 10
- 12 Machida K, Akita K, Souma Y. Development of Advanced Robotic Hand System for ETS-VII. *I-SAIRAS97.* 1997:63 ~ 68
- 13 Machida K, Nishida H, Akita K. Precise Telerobotic System for Space Experiment on ETS-VII. *49th International Astronautical Congress /Melbourne, Australia. Sept28-Oct1,1998:1 ~ 10*
- 14 Yamada K, Tsuchiya K. Efficient Computation Algorithm for Manipulator Control of a Space Robot. *Soc. Instrument Contr. Eng. Trans.* 1990, 26(7): 765 ~ 772
- 15 Yoshida K. Experimental Study on the Dynamics and Control of a Space Robot with Experimental Free-Floating Robot Satellite(EFFORTS) Simulators. *Advanced Robotics.* 1995,9(6):583 ~ 602
- 16 Yoshida K, Kurazume R, Umetani Y. Torque Optimization Control in Space Robots with a Redundant Arm. *IEEE/RSJ Int. Workshop on Intelligent Robots and Systems*, 1991:1647 ~ 1652
- 17 Yoshida K, Umetani Y. Control of space free-flying robot. *Proc. IEEE Conf Decis Control90,1990, 97 ~ 102*
- 18 Ullman MA. Experiments in Autonomous Navigation and Control of Multi-Manipulator Free-Flying Space Robots. PhD thesis, Stanford University, Stanford, July 1991
- 19 Ullman MA, Cannon RH. Experiments in Autonomous Navigation and Control of a Multi-Manipulator, Free-Flying Space Robot. *Space Robotics; Dynamics and Control*, MA:Kluwer Academic, 1991:269 ~ 284
- 20 McBarron II. Past, Present, and Future: the US EVA Program. *Acta Astronautica.* 1994,32(1):5 ~ 14
- 21 Oda M. System Engineering Approach in Designing the Teleoperation System of the ETS-VII Robot Experiment Satellite. *IEEE. Int. Conf. On Robotics and Automation*, 1997:3054 ~ 3061

- 22 Oda M. Experiences and Lessons Learned from the ETS-VII Robot Satellite. IEEE. Int. Conf. On Robotics and Automation, 2000;914 ~ 919
- 23 Oda M, Kibe K, Yamagata F. ETS-VII, Space Robot In-Orbit Experiment Satellite. IEEE. Int. Conf. On Robotics and Automation, 1996;739 ~ 744
- 24 Inaba N, Oda M. Autonomous Satellite Capture by a Space Robot. IEEE. Int. Conf. On Robotics and Automation, 2000;1169 ~ 1174
- 25 Mukherjee R, Chen D. Control of Free-Flying Underactuated Space Manipulators to Equilibrium Manifolds. IEEE Transactions on Robotics and Automation, 1993,9(5):561 ~ 569
- 26 Mukherjee R, Nakamura Y. Formulation and Efficient Computation of Inverse Dynamics of Space Robots. IEEE Trans. Robotics Automat, 1992, 8(3): 400 ~ 406
- 27 Dubowsky S, Durfee W, Corrigan T. A Laboratory Test Bed for Space Robotics; The VES-II. Intelligent Robots and Systems. 1995;463 ~ 475
- 28 Dubowsky S, Torres M. Path Planning for Space Manipulators to Minimize Spacecraft Attitude Disturbances. Proc. of IEEE Int. Conf. on Robotics and Automation. 1991;2522 ~ 2528
- 29 Agrawal SK, Shelly MP. Dynamic Simulation and Choice of Generalized Coordinates for a Free-Floating Closed-Chain Planar Manipulator. Mech. Mach. Theory. 1993,28(5):615 ~ 624
- 30 Agrawal SK, Chen MY, Annapragada M. Modeling and Simulation of Assembly in a Free-Floating Work Environment by a Free-Floating Robot. Transactions of the ASME Journal of Mechanical Design. 1996,118;115 ~ 120
- 31 Saha SK. A Unified Approach to Space Robot Kinematics. IEEE Trans. Robotics Automat, 1996,12(13):401 ~ 405
- 32 Sagara S. et al. Adaptive RMRC for Cooperative Manipulation of A Floating Object by Two Free-Based Space Robots, Proceeding of the IEEE International Conference on Intelligent Robots and Systems, 1998;1467 ~ 1472
- 33 Komatsu T et al. The Development of an Autonomous Space Robot Operation Testbed; ASROT, Advanced Robotics. 1994,8(1):165 ~ 204
- 34 Chen VW. Experiments in Adaptive Control of Multiple Cooperating Manipulators on a Free-flying Space Robot. PhD thesis, Stanford University, Stanford, March 1993
- 35 Dickson WC. Experiments in Cooperative Manipulation of Objects by Free-flying Robot Teams. PhD thesis, Stanford University, Stanford, December 1993
- 36 Dickson WC, Cannon RH. Experimental Results of Two Free-Flying Robots Capturing and Manipulating a Free-Flying Object. Proc. of the IEEE Intel. Conference on Intelligent Robots and Systems, 1995;51 ~ 58
- 37 Chen Y, Watson LT. Optimal Trajectory Planning for a Space Robot Docking with a Moving Target via Homotopy Algorithms. Journal of Robotic Systems. 1995,12(8):531 ~ 540
- 38 Nakamura Y, Mukherjee R. Exploiting Nonholonomic Redundancy of Free-Flying Space Robots. IEEE Trans. Robotics Automat, 1993,9(4):499 ~ 506
- 39 Hu YR. Motion and Force Control of Coordinated robots. PhD thesis, University of Toronto. Canada. 1991; 54 ~ 69
- 40 Hu YR, Vukovich G. Dynamic Control of Free-Floating Coordinated Space Robots, Journal of Robotic Systems, 1998,15(4):917 ~ 930
- 41 Hu YR, Vukovich G. Dynamic Control of Free-Flying Coordinated Space Robots, Proceeding of the 34th Conference on Decision and Control, New Orleans, LA-December 1995;3898 ~ 3906
- 42 Xu YS, Brown B et al. Mobility and Manipulation of a Light -Weight Space Robot. Robotics and Autonomous System. 1994, (13):1 ~ 12
- 43 Xu YS, Gu YL. Robust Control of Free-Floating Space Robot System. Int. Journal of Control. 1995,11(2): 261 ~ 277
- 44 Umetani Y, Yoshida K. Experimental Study of a Two Dimensional Free-Flying Robot Satellite Model. Proc. of

- NASA Conf. on Space Telerobotics, 1989, 5:215 ~ 224
- 45 Umetani Y, Yoshida K. Resolved Motion Rate Control of Space Manipulators with Generalized Jacobian Matrix. IEEE Trans. Robotics Automat. 1989, 5(3):303 ~ 314
- 46 Vafa Z, Dubowsky S. On the Dynamics of Manipulators in Space Using the Virtual Manipulator Approach. IEEE Proc. Robotics. Automat, 1987:579 ~ 585
- 47 Vafa Z, Dubowsky S. The Kinematics and Dynamics of Space Manipulators: The Virtual Manipulator Approach. International Journal of Robotics Research. 1990, 9(4):3 ~ 21
- 48 Murotsu Y, Senda K. Parameter identification of unknown object handled by free-flying space robot. Journal of Guidance, Control, and Dynamics. 1994, 17(3): 488 ~ 494
- 49 Yokokohji Y, Toyoshima T, Yoshikaka T. Efficient Computational Algorithms for Trajectory Control of Free-Flying Space Robots with Multiple Arms. IEEE Trans. Robotics Automat. 1993, 9(5):571 ~ 579
- 50 蔡自兴. 机器人学. 北京:清华大学出版社, 2000
- 51 陈力, 刘延柱. 漂浮基空间机械臂惯性轨迹跟踪的分解运动增广自适应控制算法. 见:空间机器人及遥科学研讨会论文集. 烟台, 1999:55 ~ 62
- 52 丁希仑, 张启先. 自由漂浮的空间机器人系统运动学及其运动规划. 见:空间机器人及遥科学研讨会论文集. 烟台, 1999:75 ~ 83
- 53 高海波, 朱利民, 熊有伦. 空间机器人系统的简化动力学模型与应用研究. 华中科技大学学报, 2001, 29(3):64 ~ 66
- 54 高子坤, 陈力. 动力学奇异点在漂浮基双臂空间机器人姿态干扰运动规划中的应用. 空间科学技术, 2003, 23(4):307 ~ 312
- 55 何光彩. 自由飞行空间机器人姿态控制研究. 哈尔滨工业大学博士论文. 1999
- 56 何光彩, 洪炳熔. 双臂自由飞行空间机器人自主控制系统仿真. 计算机仿真, 1999, 16(2):57 ~ 59
- 57 何光彩, 洪炳熔. 基于参数估计的自由飞行空间机器人双臂协调运动规划算法. 高技术通讯, 1999, (8):6 ~ 10
- 58 何光彩, 洪炳熔, 郭恒业. 基于神经网络的自由飞行空间机器人参数辨识. 见:空间机器人及遥科学研讨会论文集. 烟台, 1999:109 ~ 113
- 59 何广平, 陆震. 被动冗余度空间机器人动力学控制. 空间科学学报, 2001, 21(1):73 ~ 80
- 60 洪炳熔, 何光彩, 吴葳. 基于RMDM的双臂自由飞行空间机器人姿态控制算法. 哈尔滨工业大学学报, 1999, 31(2):7 ~ 10
- 61 洪炳熔, 李华忠, 郭恒业. 空间机器人共享自主控制研究. 见:空间机器人及遥科学研讨会论文集. 烟台, 1999:119 ~ 123
- 62 洪炳熔, 柳长安. ETS-Ⅶ型自由飞行空间机器人的系统设计策略及其空间实验. 见:空间机器人及遥科学研讨会论文集. 烟台, 1999:124 ~ 130
- 63 洪炳熔, 柳长安, 郭恒业. 自由飞行空间机器人地面实验平台系统设计. 机器人, 2000, 22(2):108 ~ 114
- 64 洪炳熔, 柳长安, 李华忠. 自由飞行空间机器人捕捉运动目标的力矩控制算法及其仿真. 宇航学报, 2000, 21(4):64 ~ 70
- 65 洪炳熔, 柳长安, 王鸿鹏. 自由飞行空间机器人概念性地面试验平台研究. 见:首届全国军事航天学术研讨会论文集. 北京, 2001:299 ~ 303
- 66 洪炳熔, 王鸿鹏, 李华忠. 空间机器人的三维几何建模及其仿真. 见:空间机器人及遥科学研讨会论文集. 烟台, 1999:131 ~ 135
- 67 洪炳熔, 王鸿鹏, 吴葳. 双臂自由飞行空间机器人的姿态控制方案. 宇航学报, 2000, 21(2):22 ~ 28
- 68 黄献龙, 梁斌, 吴宏鑫. 舱外自由移动机器人规划和运动控制. 见:空间机器人及遥科学研讨会论文集. 烟台, 1999:136 ~ 142
- 69 金飞虎, 洪炳熔, 高庆吉. 基于蚁群算法的自由飞行空间机器人路径规划. 机器人, 2002, 24(6):526 ~ 529
- 70 金飞虎, 洪炳熔, 柳长安. 双臂自由飞行空间机器人捕捉目标实验研究. 宇航学报, 2002, 23(1):64 ~ 68
- 71 金飞虎, 洪炳熔, 柳长安. 自由飞行空间机器人地面实验平台网络系统. 计算机应用研究, 2002, (8):119 ~ 121
- 72 李华忠. 基于虚拟现实的空间机器人共享控制系统. 哈尔滨工业大学博士论文. 1999

- 73 李华忠,洪炳熔. 基于 VR 的自由飞行空间机器人自主运动规划仿真系统. 计算机工程与应用, 1999,(5):1~3
- 74 李华忠,洪炳熔,郭恒业. 空间机器人的关节驱动力矩递推算法及其捕捉目标研究. 见:空间机器人及遥科学研讨会论文集. 烟台, 1999:165~169
- 75 李华忠,洪炳熔,贺怀清. 自由飞行空间机器人的关节驱动力矩递推算法及其仿真. 计算机研究与发展, 1999, 36(9):1126~1132
- 76 李华忠,洪炳熔,唐好选. 基于关节驱动力矩的自由飞行空间机器人捕捉目标控制算法. 机器人, 2000,22(3):161~168
- 77 李华忠,杨维萍,柳长安. 基于虚拟现实的空间机器人共享控制系统及其仿真. 宇航学报, 2000,21(3):100~105
- 78 连广宇,孙增圻等. ETS-VII 的交会对接和空间机器人试验. 见:空间机器人及遥科学研讨会论文集. 烟台, 1999:176~182
- 79 连广宇,孙增圻等. ETS-VII 空间机器人关键技术分析. 见:空间机器人及遥科学研讨会论文集. 烟台, 1999:170~175
- 80 梁斌,刘良栋,李庚田. 空间机器人的动力学等价机械臂. 自动化学报, 1998,24(6):761~767
- 81 刘宏. 双臂型机器人卫星运动规划研究. 哈尔滨工业大学博士论文. 1996
- 82 刘伟强. 空间机器人技术. 机器人技术与应用, 1998,(3):23~25
- 83 柳长安. 多自由飞行空间机器人协调操作研究. 哈尔滨工业大学博士论文. 2001
- 84 柳长安,李国栋,吴克河. 自由飞行空间机器人研究综述. 机器人, 2002,24(6):380~384
- 85 柳长安,洪炳熔,郭恒业. 自由飞行空间机器人地面实验平台系统策略控制. 见:空间机器人及遥科学研讨会论文集. 烟台, 1999:195~200
- 86 柳长安,洪炳熔,金飞虎. 自由飞行空间机器人地面实验平台系统规划器. 高技术通讯, 2001,(9):90~92
- 87 柳长安,洪炳熔,王鸿鹏. 自由飞行空间机器人地面实验平台硬件系统. 高技术通讯, 2001,(11):74~76
- 88 柳长安,袁景阳,李瑰贤. 多臂自由飞行空间机器人协调操作运动控制算法. 哈尔滨工业大学学报, 2004, 36(10):1338~1340
- 89 马保离,霍伟. 自由飞行空间机器人系统的协调运动控制. 自动化学报, 1998,24(1):50~55
- 90 任敬轶,孙汉旭. 自由飞行空间机器人的运动分析及仿真. 北京航空航天大学学报, 2002,28(6):613~616
- 91 曙光,洪炳熔. 基于颜色识别的自由飞行空间机器人姿态实时跟踪方法. 宇航学报, 2001,22(5):51~55
- 92 孙增圻. 空间机器人国外发展状况及我国的发展设想. 见:空间机器人及遥科学研讨会论文集. 烟台, 1999:1~20
- 93 唐好选,洪炳熔. 基于 VR 的空间机器人遥控操作仿真的研究. 计算机应用研究, 2000,(8):67~69
- 94 王鸿鹏. 基于模糊逻辑的 FFSR 智能控制研究. 哈尔滨工业大学博士论文. 2001
- 95 王鸿鹏,洪炳熔. 自由飞行空间机器人通用运动学模型及其仿真. 宇航学报, 2000,21(3):45~51
- 96 王鸿鹏,洪炳熔,郭恒业. 双臂空间机器人通用运动学路径规划算法. 系统工程与电子技术, 2000,22(7):65~68
- 97 王鸿鹏,洪炳熔,李华忠. 空间机器人的虚拟环境建模及其仿真. 计算机应用研究, 2000,(4):28~30
- 98 王景,王昊瀛,刘良栋. 自由飞行空间机器人的运动学建模研究. 航天控制, 1999,(1):6~10
- 99 王树国. 空间机器人地面试验综合平台仿真系统. 哈尔滨工业大学博士论文. 1996
- 100 吴葳. 双臂自由飞行空间机器人自主规划研究. 哈尔滨工业大学博士论文, 1998
- 101 吴葳,洪炳熔. 空间机器人的建模及仿真. 计算机工程与设计, 1998,19(6):3~7
- 102 吴为民. 单臂型机器人卫星运动规划研究. 哈尔滨工业大学博士论文. 1995
- 103 赵晓东,王树国. 障碍环境下空间机器人路径规划算法及其仿真. 高技术通讯, 2003,(6):54~58
- 104 赵晓东,王树国. 基于载体姿态调整的自由漂浮空间机器人路径规划方法的研究. 高技术通讯, 2003,(3):47~51
- 105 赵晓东,王树国. 基于轨迹规划的自由漂浮空间机器人抓取运动物体的研究. 宇航学报, 2002,23(3):48~51
- 106 郑洁,张卫民. 空间机器人的三维动画仿真. 机电一体化, 2004,(3):37~39
- 107 朱铁一. 自由飞行空间机器人地面试验平台视觉系统的研究. 哈尔滨工业大学博士论文. 1997

内 容 简 介

本书是关于自由飞行空间机器人运动控制及仿真技术方面的专著,大部分内容是作者近 10 多年来在该领域研究成果的积累与总结,部分内容还是首次发表。

本书共分十二章。第一章介绍自由飞行空间机器人的概况,重点阐述了研究意义,研究现状以及未来发展趋势。第二章 ~ 第十一章主要介绍自由飞行空间机器人运动控制的基本理论和方法,包括运动学模型及运动特性,基于神经网络的参数辨识,捕捉目标运动规划,基于受限最小干扰图的姿态控制,基于姿态稳定的冗余运动规划,姿态控制综合算法,基于关节力矩的运动控制算法,模糊运动控制算法,飞行轨迹优化算法以及多机器人协调操作控制算法等。本书第十二章自由飞行空间机器人地面实验平台,以作者所在单位研制的一台双臂自由飞行空间地面实验平台为对象,介绍地面实验系统的组成、工作原理以及实验方法。

本书可供从事空间机器人、机器人学、机器人控制等研究领域的研究人员参考。

阅读此文的还阅读了:

1. [给教师更大的自由空间](#)
2. [自由空间无法分区](#)
3. [给孩子一个自由的空间](#)
4. [给孩子自由绘画的空间](#)
5. [还孩子自由写作的空间](#)
6. [给学生一片自由的空间](#)
7. [五轴飞行仿真转台](#)
8. [成长,需要自由空间](#)
9. [虚拟空间演绎“空战”革命——记空军某飞行学院飞行仿真技术研究所](#)
10. [自由的空间](#)
11. [“自由填埋”的展示空间](#)
12. [在自主的空间里自由翱翔](#)
13. [多给孩子自由展示的空间](#)
14. [自由空间犀鸟的](#)
15. [多给孩子自由展示的空间](#)
16. [网络空间要自由更要秩序](#)
17. [某型导弹飞行轨迹的仿真](#)
18. [某型弹道导弹自由段飞行弹道仿真](#)
19. [飞行:一种神奇的自由](#)
20. [给孩子一个自由的空间](#)
21. [《自由的空间2》玻璃钢](#)
22. [给学生一片自由的空间](#)
23. [给学生自由的空间](#)
24. [给孩子自由的空间](#)
25. [空间是为自由设定的](#)
26. [还学生自由的学习空间](#)
27. [空间飞行器](#)
28. [犀鸟的自由空间](#)
29. [在空间自由移动的硬币](#)
30. [给我们一个自由的空间吧!](#)
31. [持续载荷飞行仿真梁端空间算法研究](#)

32. 给孩子一个自由的空间
33. 弹性飞行器飞行仿真研究
34. 给孩子更多的自由空间
35. 公文制作的“自由空间”
36. 自由，在适度的空间里
37. 自由的空间
38. 多给孩子自由的空间
39. 创造自由的办公空间
40. 给教师更大的自由空间
41. 犀鸟的自由空间
42. 飞行的空间(组诗)
43. 自由飞行在低空
44. 给密友自由的空间
45. 自在空间 自由生活
46. 给孩子自由的空间
47. 自由业者的自由空间
48. 给学生一片自由的空间
49. 超空间飞行
50. 期盼“自由”飞行的通用航空