

哈尔滨工程大学
硕士学位论文
智能水下机器人全局及局部路径规划技术研究
姓名：付岩
申请学位级别：硕士
专业：计算机应用技术
指导教师：顾国昌
20040201

摘 要

路径规划技术是智能水下机器人(Autonomous Underwater Vehicle, 简称 AUV)的关键技术之一,在一定程度上它标志着水下机器人智能水平的高低。本文对大范围海洋有流环境下智能水下机器人的路径规划技术进行了系统深入的研究,主要包括以下几个方面的内容:

本文首先介绍了机器人以及智能水下机器人的发展情况,分析了智能水下机器人的国内外的研究动态和发展趋势,详细介绍了课题的背景和主要内容。

其次,简要介绍了机器人路径规划技术,包括路径规划的定义、分类、相关问题、几种典型的路径规划方法以及最新的研究动态等。在分析大范围海洋环境下路径规划的复杂性以及总结以前的路径规划算法的基础上,提出了一整套从全局到局部的路径规划算法。

最后,本文分别详细介绍了全局路径规划和局部路径规划两个算法:基于分层模型的遗传模拟退火算法路径规划和基于雷达法的局部路径规划。包括算法的基本思想、算法实现中的几个关键技术等。本文还给出了两个算法的计算机仿真结果。

关键词: 智能水下机器人(AUV); 路径规划; 分层模型; 遗传模拟退火算法; 雷达法;

ABSTRACT

Path planning is one of the key technologies of AUV(Autonomous Underwater Vehicle), and it represents the intelligence level of AUV to some extent. So, this paper studies systematically the path planning for AUV in a large sea environment with ocean current. The following aspects are investigated and discussed.

Firstly, this paper introduces the development of robot and AUV, analyses the domestic and foreign research trends of AUV and introduces the background and the main content of this paper.

Secondly, the definition and classification of robot path planning, some typical path planning approaches and the new research trends of path planning are elaborated. After analyzing the complexity of path planning in a large sea environment with ocean current and the shortcoming of the previous path planning approach, this paper presents a suit of path planning approaches from global to local.

Finally, the two path planning approaches are elaborated differently, which are the Genetic Simulated Annealing path planning approach based on hierarchical model and the local path planning approach based on radar path planner. This paper elaborates the main idea, and some details of the two approaches. The simulation results of the two approaches are presented.

Keywords: AUV(Autonomous Underwater Vehicle), path planning, hierarchical model, Genetic Simulated Annealing Algorithm, radar path planner

哈尔滨工程大学

学位论文原创性声明

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

作者（签字）： 付岩

日期： 2004 年 2 月 29 日

第1章 绪 论

1.1 机器人概述

机器人是传统的机构学与近代电子技术相结合的产物，也是当代高技术发展的一个重要内容。机器人特别是智能机器人的出现标志着人类对机器与智能的探索达到了一个崭新的阶段。机器人技术集力学、机械学、电子学、生物学、控制论、计算机、人工智能、系统工程等多种学科之大成，是一种综合性很强的新技术。从第一代 1948 年诞生于美国阿贡实验室的遥控机械手发展到今天，机器人技术已经经历了三个主要阶段^[1]：

第一代机器人是主从式遥控机器人，以示教再现的方式工作，不考虑反馈因素。它们装有记忆存储器，由人将作业的各种操作要求示范给机器人，使之记住操作的程序和要领。当它接到再现命令时，则自主地模仿示教的动作行事。示教内容为机器人操作机构的空间轨迹、作业条件、作业顺序等。示教方法或是操作员“手把手”地直接做，或是与计算机编程相结合。

第二代机器人是适应性程序控制机器人，具有某些感觉装置，能获取作业环境、操作对象的简单信息，通过计算机处理、分析，机器人做出一定的推理，对动作进行反馈控制，表现出较低级的智能，比第一代机器人更灵活、更能适应环境变化的需要。

第三代机器人是智能机器人，具有多种感知功能，可以进行复杂的规划、判断与决策，能在作业环境下独立工作。它是人工智能、计算机技术与工业机器人三者相结合的产物，目前尚处于实验探索阶段。

机器人按其使用目的可分为四类：智能机器人，具有类似于人的感觉器官，能一边了解周围的状况，一边用手和脚来进行工作；工业机器人，应用在工厂生产线上以代替人的劳动；操纵机器人，靠人的远距离操纵来进行作业；福利机器人，具有可代替手或脚有缺陷的残疾人的动作功能。

虽然智能机器人是目前机器人领域的研究热点，但迄今对它也没有一个

确切的定义。长期以来,人工智能学者一直把机器人作为研究人工智能的载体,来研究如何使机器人具有环境识别、问题求解以及规划能力,祈望使机器人具有类似人的高度自治功能,这样就在很大程度上把研究局限于计算机科学的范畴,只把智能机器人看为一个纯软件系统。事实上,这种看法是不全面的。应该把智能机器人看成是一个软件、硬件组成的总体,简单地讲,智能机器人是具有感知、思维和动作的机器^[2]。所谓感知(Sensing),是指具有发现、认识和描述外部环境和自身状态的能力;所谓思维(Thinking)是说智能机器人自身具有解决问题的能力,或者它会通过学习,自己找到解决问题的办法;而所谓动作(Acting),则说明了智能机器人不是一个单纯的软件体,它具有可以完成作业的机构和驱动装置。由于人们对自身智能行为的认识还很肤浅,所以目前则不能对机器人的智能期望过高,对能在一定程度上感知环境,具有一定适应能力和解决问题本领的机器人,我们就可以称之为具有“智能”^[2]。

从应用领域来看,机器人还可分为空间开发机器人,室内移动机器人,室外移动机器人,水下机器人和特种机器人等。

移动机器人的研究始于 60 年代末期。斯坦福研究院(SRI)的 Nils Nilsson 和 Charles Rosen 等人,在 1966 年至 1972 年中研造出了取名 Shakey^[3] 的自主移动机器人。目的是研究应用人工智能技术,在复杂环境下机器人系统的自主推理、规划和控制。与此同时,最早的操作式步行机器人也研制成功,从而开始了机器人步行机构方面的研究,以解决机器人在不平整地域内的运动问题,设计并研制出了多足步行机器人。其中最著名是名为 General Electric QuadNped^[4] 的步行机器人。70 年代末,随着计算机的应用和传感技术的发展,移动机器人研究又出现了新的高潮,特别是在 80 年代中期,设计和制造机器人的浪潮席卷全世界,一大批世界著名的公司开始研制移动机器人平台,这些移动机器人主要作为大学实验室及研究机构的移动机器人实验平台,从而促进了移动机器人学多种研究方向的出现。90 年代以来,以研制高水平的环境信息传感器和信息处理技术,高适应性的移动机器人控制技术,真实环境下的规划技术为标志,开展了移动机器人更高层次的研究。

1.2 智能水下机器人 (AUV) 概述

1.2.1 AUV 的产生背景和研究意义

随着陆地资源的不断消耗,海洋逐渐成为人类赖以生存的第二空间。海洋地位的提高,使得争夺海洋岛屿、海洋资源、海洋国土的斗争交织在一起,形成一系列海上热点,海洋成为大国争夺的主战场,沿海国家也纷纷把国家外交与国防的重点转向海洋。我国是一个濒临海洋的国家,毗邻我国大陆边缘的海区总面积达 300 多万平方公里。

由于常规的海洋舰船和武器系统在应付现代高科技海洋战争中具有一定的局限性,因此,世界各先进国家都纷纷开始研究各种智能水下机器人,利用其隐蔽性和长期性等特点,执行各种特殊的军事使命(预警、侦察、布雷和反水雷等)。

水下机器人技术是多学科高技术的集成体,涉及光、机、电、声学以及自动控制、计算机技术、流体力学、密封技术等等。水下机器人根据不同的特征可以有不同的分类。按其在水中运动的方式可以分为浮游式水下机器人、步行式水下机器人和移动式水下机器人三种^[1]。而按照控制方式又可以分为有缆式水下机器人和无缆式水下机器人两种。而智能水下机器人(Autonomous Underwater Vehicle, 以下简称 AUV)是水下机器人的一个重要组成部分。它拥有多种感觉装置,具有自主导航和避障能力,并能自主地完成某些特定的水下作业任务。它是含机械学、电子学、自动控制、模式识别、计算机及人工智能等多种高技术的集成体。该领域的专家将体系结构、任务规划、定位和导航、自主式作业、用户与机器人系统的接口、知识表示和获取、传感器及传感器处理、三维图像感知等 11 个方面列为当前研究的重点和热点。本文所要研究的是无缆的水下浮游式智能机器人。

AUV 在海洋科学考察和海洋工程项目支持保障方面发挥非常重要的作用,AUV 可以用于完成深海建模任务,即对全球范围海洋活动进行更广泛的三维观测;AUV 可以用于海底探测以便评价海洋资源是否有工业开采价值。AUV 还可以当作 450 米以下的深水工作平台使用。在潜水员能够工作的浅水

区域, AUV 可以当作潜水员的辅助工具以提高这种操作的效率, 并且提高此类操作的安全性和便于对紧急事件的处理。AUV 还可以用来执行水下调查任务, 如用海洋科学考察船货载人或无人电缆 ROV 系统进行此项工作, 诸如水下管道和电缆这样的水下调查任务和水下摄影任务, 耗资将是非常可观的。据估计, AUV 进行水文调查的工作效率是海洋考察船的六倍。若用 AUV 执行现有海洋科学和工程系统的保障任务, 将会大大提高系统的效率; 而且在取代现有许多载人或无人有缆 ROV 方面有着巨大潜力, 在经济上也是十分合算的。所以 AUV 无论在海洋科学研究还是在国防及经济等方面都有重要价值和特殊的意义。

1.2.2 AUV 技术的研究发展

目前, 各海洋大国的许多大学和研究机构都投入了大量的人力和资金, 用于研究和开发智能水下机器人。在 AUV 的开发过程中, 美国、加拿大、英国、法国、俄罗斯、德国、意大利等国起步较早。

50 年代末期, 美国华盛顿大学开始建造第一艘无缆水下机器人——SPURV, 华盛顿大学物理实验室研制了两台样机, 称为 SPURV I 和 SPURV II, 主要用于海洋研究中的某些参数测量, 测量结果用磁带记录仪加以记录。60 年代中期, 产业界和军方开始对无缆水下机器人发生兴趣。但是, 由于当时技术上的难度太大, 产业界和军界又将兴趣转移到载人潜水器上。此后, 无缆水下机器人由于技术上的原因在低水平上徘徊了多年。70 年代中期, 由于微电子技术、计算机技术、人工智能技术、导航技术的飞速进展, 再加上海洋工程和军事活动的需要, 国外产业界和军方再次对无缆水下机器人发生了兴趣^[5]。

在本世纪最近的 20 多年内, AUV 得到了很大的发展, 开发了一批能工作在各种不同深度、进行多种作业的机器人。AUV 代表了未来水下机器人技术的发展方向, 是当前世界各国研究工作的热点^[5]。

1. 国外 AUV 技术的研究发展

美国的 AUV 研究机构多, 技术力量也非常雄厚, 代表性的 AUV 有美国海军研究生院 (Naval Postgraduate School) 研制的 “Phoenix”、新罕布什尔

大学(University of New Hampshire)海洋系统工程实验室研制的“EAVE III”、麻省理工学院(Massachusetts Institute of Technology)研制的“Odyssey II”、佛罗里达大西洋大学(Florida Atlantic Technology)研制的“Ocean Voyager II”等。

其中New Hampshir大学研制出的EAVE-EAST自主式水下机器人基于知识的系统分为高级与低级两部分,高级部分处理的内容包括:数据评价、状态评价、监控和引导;低级部分包括实施管理、传感器管理和系统监视。1987年10月该系统在美国winnipesavkee湖中演示了进行水下侦察的情况。这次演示使用两台无缆水下机器人,彼此可以进行编队航行,侦察时进行动作协调以躲避搜索,EAVE-EAST实现了人工智能导航定位和避障试验^[6]。

在军用方面,美国海军正下大力研制自治潜水器AUV。美国海军制定了一项雄心勃勃的AUV发展计划,打算首先突破水雷侦察这一关。自治潜水器将由潜艇发射,它将成为潜艇的制式装备。该计划包括两个项目,一个是近期水雷侦察系统NMRS,另一个是长期水雷侦察系统LMRS。1994年8月,美国海军与诺斯洛普·格鲁曼公司签订了研制合同。海军要求1998年提供一个NMRS的作战样机系统并完成实验,它应具备初步作战能力,供舰队使用,2000年可参加作战。该系统将由SSN—688级潜艇发射及回收,可以对水雷进行探测、分类及定位^[7]。

日本的AUV研究目前处于世界的领先地位,他们在研究开发的同时,还进一步强调AUV技术的实用化。东京大学URA(浦)实验室开发的多种能力不同的自主式水下机器人,如“Twin-Burger”、“PTEROA150、250”及“R1”等。

英国工业和贸易部的海洋资源开发计划提供资金,马可尼水下系统有限公司、Moog控制设备公司以及Chelsea仪器公司在重型鱼雷的基础上制造了用于北冰洋冰下勘测的AUV。该AUV将配备有上视和侧扫声纳以及一个带有磁通量闸门罗盘和电磁式测程仪的三轴惯性导航系统。英国国防部正为国家环境研究委员会的自主式水下航行器(AUTOSUB)研究计划提供资金,以便评估军用AUV的潜力以及需要解决的工程和技术难题。海洋科学研究所Deacon实验室对AUTOSUB的研究负全部责任,该项目从1988年已开始实施。AUTOSUB计划的主要目标是研制两种远距离航行的AUV。这些AUV

在越洋航行时能连续采集海洋表面和海床间的数据,并自主地遥感深海底部。这两艘 AUV 分别叫“小狗”(DOGGIE,恰是“深海和地球物理测量艇”的缩写)和海豚(DOLPHIN,恰是“深海远程水文地理测量艇”的缩写)^[8]。

在加拿大,在过去载人有缆和无人有缆遥控潜水器开发的基础上,AUV 的技术也得到了迅速发展,开发自主式潜水器的主要任务在于降低诸如水下调查这样的作业任务费用,DOLPHIN 和 ARCS 是两台典型的 AUV。DOLPHIN 是一台以无线电遥控传递信息的 AUV;而 ARCS 自主遥控潜水器则是一种用于冰下调查、监视和侦察的潜水器,本体设有分布式微处理机结构以便执行多种软件任务。

法国在 1981 年开发了一台用于海底摄影和海底地形测试的自主式潜水器 EPAULARD。最近又对 EPAULARD 进行了重装,其目的是使该系统能够在起伏不变的海底地形上进行作业,EPAULARD 在 2300m 水深进行了大量摄影工作,而且这些图像都成功地传到了水面母船。

俄罗斯也相当重视自主式潜水器技术发展。1975 年到 1976 年他们制造了两台类似的潜水器 PISCES VI 和 PISCES XII,在各大洋十年的深水作业已经证明这两台潜水器是功能最全的、推力最大的研究工具。自此以后,他们又设计和制造了两台潜水器 MIR1 和 MIR2,两个潜水器的试验工作是在大西洋中部水域完成的。在 1990 年,科雷洛夫造船研究所开发出一台无人潜水器 URAN-1,其额定潜深为 6000m。在 1988 年,俄罗斯科学院远东海洋技术问题研究所又开发出一台用于水下调查和观察的无人潜水器 MAKS-2 和用于海深搜索和调查的 SEA LION^[9]。

其它经济较为发达的海洋国家或地区,也在进行 AUV 的研究开发,如韩国的“VORAM”,台湾地区的“AUV—HM1”等。

2. 国内 AUV 技术的研究发展

我国研究 AUV 技术的起步较晚,八十年代初以前,我国在水下机器人技术方面几乎是一片空白。1982 年中国科学院将试制“海人一号”列为院重点项目。试制“海人一号”的目标是探索这一领域的基本学科原理和验证单元技术。沈阳自动化所利用本所的技术储备,并与上海交大等单位多方合作,四年中投入人力近 300 人,突破了关键性总体设计技术,有十几项技术属国内首创。1985 年 12 月“海人一号”在大连海域首航试验获得成功,以后又

多次试验,证明达到海军提出的综合指标,下潜 200 米抬起指定目标物。“海人一号”是我国第一台自行研制成功的水下机器人样机,在我国发展水下机器人事业中起到奠基石作用^[10]。

在 1987 年,国家人工智能专家组提出了关于研制一台无人无缆自主式潜水器试验床作为中国 AUV 技术开发的关键性起步的建议,并把 AUV 开发计划列入国家“863”自动化领域的研究项目中。七五期间,国家将“海洋和水下机器人技术开发”列为重点科技攻关课题。沈阳自动化所承担了中型水下机器人产品的攻关任务,在引进美国的 RECNO-IV 技术的基础上,突破关键技术,进行创新设计,在七五期间完成了三台中型水下机器人,有两台打入国际市场,并获得了一致好评。

目前,各种机器人的生产和研制在发达国家方兴未艾,而在我国,无论是开发应用方面还是学术研究方面都竞相争上,一个立足于国情,跟踪世界先进水平的“机器人热”正在形成。

我国在水下机器人的研究上,民用方面以“863”项目为代表。中科院沈阳自动化所与哈尔滨工程大学等单位共同研制开发了工作水深 1000 米的“探索者”号无缆水下机器人,1994 年完成海试。后来中科院沈阳自动化研究所又通过与俄罗斯合作,共同开发了观察型、预编程控制的“CR—01”无缆水下机器人,工作水深为 6000 米^[11]。该机器人完成了太平洋深海的考察工作,达到了实用水平。

另外,由哈尔滨工程大学等承担的原国防科工委跨部门预研重点项目,其“八五”计划经过水池与湖中实验,研制成功机器人试验平台“XX—I”和“XX—II”型。

“XX 智能水下机器人”项目作为八五和九五期间的重点科研攻关项目,其总体目标是研究水下智能机器人技术,研究和设计名为“XX—III” AUV。该 AUV 是一浮游式潜水器,其长度为 4.0 米,直径为 0.8 米,其上配有避碰声纳、短基线定位声纳、深度计、电罗经、摄像机、成像声纳和机械手等设备。系统在八五期间的目标是使试验平台具有演示三项智能行为的能力,即:

- (1) 自主导航能力:根据已知地图,在水池和湖中从不同起始点到不同终点的自主导航能力。

- (2) 自主避障能力：当航行过程中出现突发障碍物时，机器人应具有检测和绕过障碍物的能力。
- (3) 自主简单作业能力：用剪缆和投放等方式，模拟完成剪断锚雷链与消除沉底雷等任务。

该系统在九五期间的目标是进一步提高对环境的适应能力和智能水平，要求在海洋环境自主完成绘制雷区图的作业及自主完成模拟援潜救生的部分作业^[12]。该系统于 1995 年完成了水池实验和湖试，于 2000 年 6 月完成了海洋试验，并于 2000 年 10 月通过了国防科工委主持的鉴定。参加测试的专家们认为：在真实的复杂海洋环境中，搜索和识别真正的水下目标，这是 XX 智能水下机器人向实用化迈出的关键一步，证明了“XX—III”型水下机器人在自主性和适应性两个方面都有一定的突破，智能水平又向前迈进了一大步。

十五期间，国防科工委再次将“XX 智能水下机器人技术”作为重点预研项目，下节将就该项目的情况加以介绍。

3. AUV 的发展前景

AUV 代表了未来水下机器人技术的发展方向，是当前世界各国研究工作的热点。当前 AUV 的发展趋势为更深、更远、功能更强大，特别是未来海上作战等军事需求的增加，给 AUV 的发展带来了无限生机，也预示着 AUV 开始走向应用阶段。目前，AUV 的发展趋势主要表现在以下几个方面：

(1) 向远程发展

AUV 的分类方法有几种，其中一种是按照航程的远近分为远程和近程两类。所谓远程是指 AUV 一次补充能源连续航行超过 100 海里以上，而小于 100 海里称为近程。

远程 AUV 涉及的关键技术包括能源技术、远程导航技术和实时通信技术。因此，许多研究机构都在开展上述关键技术的研究工作，以期获得突破性的进展。也只有在上述关键技术解决后，才能保证远程 AUV 计划的实施。目前正在研究的各种可利用的能源系统包括一次电池、二次电池、燃料电池、热机及核能源。开发利用太阳能的 AUV 是引人注目的新进展，太阳能 AUV 需要浮到水面给机载能源系统再充电，并且这种可利用的能源又是无限的。

(2) 向深海发展

地球上 97% 的海洋深度在 6000 米以上，称之为深海。研制 6000 米的潜

水器是许多国家的目标。虽然无人有缆遥控水下机器人(ROV)和载人潜器也达到了这个深度,但是由于 AUV 与载人潜水器相比较,它具有安全(无人)、结构简单、重量轻、尺寸小、造价低等优点;与 ROV 相比,它具有活动范围大、潜水深度深、不怕电缆缠绕、可进入复杂结构中、不需要庞大水面支持、占用甲板面积小和成本低等优点,AUV 深海发展对海洋研究与开发更为有利。法国、美国、俄罗斯、中国都先后研制了 6000 米级的 AUV。

(3) 向功能更强大的方向发展

这里所说的功能强大主要指两个方面。一个是指 AUV 具有较强的作业能力,另一个是指 AUV 具有更高的智能。

目前所开发的 AUV 绝大多数只能用于观察和测量,不具备作业能力,使 AUV 具有作业能力是一个长远的目标。为了使 AUV 有作业能力,要在 AUV 上安装作业工具包,其中最有代表性的作业工具是水下机械手。美国麻省理工学院、卡内基梅隆大学、夏威夷大学以及法国和意大利的一些研究所和公司,正在从事这方面的工作。其中意大利的水下监控机械手已达到实用水平。

增加水下机器人智能行为一直是各国科学家的努力目标。但是由于目前的人工智能技术不能满足水下机器人增强智能的需要,因此需要将人的智力引入到水下机器人中来,这就是监控型水下机器人的思想。不完全依赖于机器的智能,更多地依赖传感器和人的智能,是今后的一个重要发展方向。我们把这种机器人称为基于传感器的先进水下机器人。发展多机器人协同控制技术,也是增加 AUV 智能的重要方面。

(4) 发展军用 AUV

AUV 发展的最大动力来自于军事需要,前述的几个发展趋势也是和军用目的联系在一起的,未来的海战有可能在全球的海岸区发生,建立海岸安全操作区,开发海洋侦察系统和先进的水雷对抗系统已被提上了日程。研制由潜艇、直升机、水面战舰或其他手段(如民船)发射的 AUV 是这些系统的核心内容。

此外,发展极浅水区用的 AUV 和经济型 AUV 也是值得注意的动向。在极浅水区,如 6 米以浅的区域,支持母船无法进入,因此依赖母船的机器人系统不能发挥作用,这种场合 AUV 则能发挥作用。发展低成本的经济型 AUV

有利于 AUV 的推广应用,也是重要方向。

总的来说, AUV 技术成熟到满足商业化应用还需要一段时间,至少需要五年以上,但是 AUV 未来的应用前景是极其广阔的。

1.3 课题的背景和主要内容

本文所研究的课题来源于“XX 智能水下机器人技术”项目。该项目的总体目标是在九五的基础上,继续研究水下机器人智能技术与实用化及关键性的工程实现技术,构成以潜艇(或水面舰艇)为基地,作战半径不低于 20km,在航道、港湾、登陆区域等重要区域进行水下侦察(探雷、搜索水下障碍物等),以及在更大范围进行远程精确布雷等任务为背景,并能在相应的海洋环境下进行演示的、基本可用的水下智能机器人样机系统(XX—IV),为尽快成为海军装备奠定坚实的技术基础。项目的主要研究内容包括:研究水下智能机器人系统的仿真技术、水下目标的探测与识别技术、水下导航与定位技术、智能控制技术、信息融合与理解及智能化作业技术、系统集成技术^[13]。

本课题组承担 XX 水下智能机器人项目中“智能控制技术”和“系统集成技术”中的部分工作,研究的主要内容及关键技术包括:

- (1) 用于海洋环境下,受机器人能力制约及环境、任务等约束的全局路径规划和局部路径规划技术研究。
- (2) 自主导航和避碰技术的研究。研究对环境的感知学习方法,在海洋环境下和未知地图的路径规划技术,各种避碰策略和方法研究。
- (3) 水下智能机器人的软硬件体系结构。根据“十五”确定的总体目标,XX 智能水下机器人软硬件体系应满足全部在机器人本体内的嵌入式安装,同时应便于控制软件的修改、调试和下载,以及满足系统的小型化和可靠性的要求。
- (4) 远程精确布雷和自主水下侦察技术的研究。

1.4 作者的主要工作和论文的组织

作为本课题组的一位成员,作者主要承担并完成了以下工作:设计出一

套适合于在大范围海洋环境下智能水下机器人的全局路径规划和局部路径规划方法，其中要求全局路径规划方法能够很好的利用内存空间，较好的解决大范围路径规划问题中遇到的求解精度与内存空间的矛盾；同时要求局部路径规划方法既要满足规划精度和速度的要求又要考虑到机器人的姿态以及海流对机器人行走的影响；同时，在对水下机器人路径规划方法的理论研究和实际应用过程中，成功的解决了原有算法出现的问题，并对原有算法在一定程度上进行了改进和提高。

本文是根据作者所做的工作进行组织安排的，具体组织方式如下：

第一章简要地介绍了智能水下机器人（AUV）的有关内容，包括研究的意义，发展动态及本课题的背景和内容等，并提到了作者的工作和论文的组织。

第二章介绍了机器人路径规划技术，包括路径规划的定义、分类、相关问题、几种典型的路径规划方法以及最新的研究动态等，并给出了本文提出的路径规划方法的基本思想。

第三章阐述了全局路径规划方法的实现方案——基于分层模型的遗传模拟退火算法路径规划。

第四章阐述了局部路径规划方法的实现方案——基于雷达法的局部路径规划。

最后，作者对所做的工作加以总结，并提出了有待进一步探讨的问题。

第 2 章 机器人路径规划技术

2.1 路径规划概述

随着机器人在工业领域的应用越来越广泛,人们对机器人的“智能”的要求也在提高,要求机器人在运动过程中具有与周围环境交互的能力。其中一个最基本的需求就是能规划出避碰的运动路径,这种需求即为路径规划问题。路径规划技术是智能机器人领域中的核心问题之一,也是机器人学中研究人工智能问题的一个重要方面,在一定程度上标志着水下机器人智能水平的高低。

2.1.1 路径规划问题的描述

移动机器人的导航问题主要是由 Durrant-Whyte H F 提出的三个问题^[14](1)“我现在何处?”(2)“我要往何处去?”(3)“要如何到该处去?”路径规划是移动机器人导航中的最重要的任务之一,它被描述成:给定一个移动机器人所处的环境(环境可以通过移动机器人视觉系统或其他途径获得),一个起始点和一个期望的目标点,机器人路径规划根据一定的任务要求,寻求一条连接起始点到终止点且能避开环境中障碍物的运动轨迹,即最优或次优有效路径。蒋新松在文献[2]中为路径规划做出了这样的定义:路径规划是自治式移动机器人的一个重要组成部分,它的任务就是在具有障碍物的环境内,按照一定的评价标准,寻找一条从起始状态(包括位置和姿态)到达目标状态(位置和姿态)的无碰路径。

J. T. Schwartz 和 M. Shair^[15] 对机器人的运动规划是这样描述的:设 R 是一个机器人系统,它是由一系列刚体性质的部件互相连接(或不连接,即多机器人系统)而成,总共有 k 个自由度。假设 R 可在 2 维或 3 维的工作空间 V 中,在不破坏自身运动约束的情况下自由运动;同时,在工作空间中有一系列几何参数已知的障碍。路径规划是指对给定的一个初始位形 Z_i 和一个目

标位形 Z_2 (位形是指能完全确定机器人系统状态的机器人的位置和姿态), 要求求解一个机器人系统 R 的连续动作, 使之无碰地从 Z_1 运动到 Z_2 。

需要指出的是, 路径规划 (Path Planning) 不同于轨迹规划 (Trajectory Planning), 路径规划一般只给出机器人在几何意义上的位置和转角, 而轨迹规划同时还要考虑机器人的线速度和角速度。因此, 路径规划问题可以看作是轨迹规划问题的一个子集, 它忽略了机器人的动力学特性。

机器人路径规划问题在理论上主要存在三个子问题^[1]:

(1) 环境表示问题: 指环境中障碍物的表示和自由空间的表示。环境建模就是对机器人活动空间的有效描述。机器人在规划前首先要做的就是将环境的描述由外部的原始形式通过一系列处理转化为适合规划的内部的世界模型, 这个过程我们称为环境建模, 其中主要是障碍物的表示方法。合理的环境表示才能有利于规划中搜索量的减少, 才能有利于时空开销的减少。不同的规划方法是基于各种不同的环境建模来进行的。

(2) 寻空间问题: 在某个指定区域 R 中, 确定物体 A 的安全位置, 使它与已有的其它物体 B_j ($j=1,2,\dots,m$) 相碰。

(3) 寻路径问题: 在某个指定区域 R 中, 确定物体 A 从初始位置移动到目标位置的安全路径, 使得移动过程中不发生 A 与 B_j 的碰撞。一系列安全位置就可以构成安全路径。

解决路径规划问题的关键在于设计一种恰当的环境模型表示方法, 即一种数据结构。该结构能反映出机器人在某位置上发生碰撞的可能性。然后, 按照模型中的连通性构造与之对应的可连通图。该连通图仅抽取环境模型的连通性, 忽略掉与路径规划不相干的信息, 避免了因无关信息的引入使规划复杂化, 这种连通图构成了问题状态空间。在该状态空间内利用搜索技术进行搜索, 便求解出一条通路。将通路上的节点转化为实际模型中的表示形式 (如坐标点的序列), 便得到了机器人的无碰路径。

2.1.2 路径规划的分类及特点

大多数导航系统由两级规划组成, 即局部规划 (Local Planning) 和全局规划 (Global Planning)。前者主要解决机器人定位和路径跟踪问题, 而后者主要

解决全局目标分解成局部目标，再由局部规划实现局部目标。根据环境信息的完整程度，移动机器人路径规划问题可分为两种^[16]：一种是基于先验完整信息的全局路径规划，又称静态规划；另一种是基于传感器信息的局部路径规划，又称动态规划，后者环境是未知或部分未知的。

路径规划问题已有的研究方法可以分为全局型方法、局部型方法以及混合型方法三种。

全局规划方法依照已获取的环境信息，给机器人规划出一条路径。规划路径的精确程度取决于获取环境信息的准确程度。全局方法通常可以寻找最优解，但是需要预先知道环境的准确信息，并且计算量很大。

局部规划方法侧重于考虑机器人当前的局部环境信息，让机器人具有良好的避碰能力。很多机器人导航方法通常是局部的方法，因为它的信息获取仅仅依靠传感器系统获取的信息，并且随着环境的变化实时的发生变化。和全局规划方法向比较，局部规划方法更具有实时性和实用性。缺陷是仅仅依靠局部信息，有时会产生局部极点，无法保证机器人能顺利到达目的地。

混合型方法试图结合全局和局部的优点，将全局规划的“粗”路径作为局部规划的子目标。从而引导机器人最终找到目标点。

路径规划问题的研究涉及到环境表达、规划方法、路径执行。环境表达有两层含义，其一是有效的获取环境信息，这与视觉/传感器相关；其二是如何将环境信息有效地表达出来。规划方法关心的是在环境表达的基础上，采用有效的方法规划路径并且进行优化。路径的执行与底层控制密切相关，并且考虑机器人的动力学特性，控制机器人按照设定的路径行走。

按照障碍物的稀疏可把路径规划问题划分成粗规划和细规划。前者考虑问题时所涉及到的自由空间远大于机器人的尺寸与机器人定位误差的和。后者考虑在狭窄空间下的规划问题，它所要求的移动精度高于机器人定位误差的精度。

按照环境的特点，路径规划问题可分为“时变”规划问题和“时不变”规划问题。当环境或障碍物的位置可以发生改变时，称之为“时变”规划问题，否则称为“时不变”规划问题。如果机器人可以移动一部分障碍物，又称为“可移障碍物”规划问题。

另外，根据完整性还可以把路径规划算法分为确定型和启发型两类。确

定型算法可以找到一个问题的答案，除非它证明该问题无解。启发型算法致力于在较短的时间内找出问题的答案，它可能无法解决较困难的问题或无法给出一个问题的最优解。

路径规划问题具有如下特点：

(1)复杂性：在复杂环境尤其是动态时变环境中，机器人路径规划非常复杂，且需要很大的计算量。

(2)随机性：复杂环境的变化往往存在很多随机性和不确定因素。动态障碍物的出现也带有随机性。

(3)多约束：机器人的运动存在几何约束和物理约束。几何约束是指机器人的形状制约，而物理约束是指机器人的速度和能量等。

(4)多目标：机器人运动过程中路径性能要求存在多种目标，如路径最短，时间最优，安全性能最好，能源消耗最小。但这些目标之间往往存在冲突。

2.2 几种路径规划方法

移动机器人路径规划是移动机器人技术的主要研究内容之一，已经有近四十年的研究工作，即从不同的环境建模，针对不同的应用领域等提出了很多路径规划的方法，几何法、单元分解法、人工势场法和数学分析法就是路径规划中比较基本的方法。随着路径规划问题的越来越复杂，遗传算法、神经网络等人工智能的方法也因此被引入到路径规划问题当中来。多种方法不是互相排斥的，常常是几种方法结合起来，互相取长补短，共同地实现路径规划任务。下面就由浅入深地介绍一下路径规划的几种方法。

2.2.1 位姿空间法

位姿空间法是目前研究移动机器人路径规划的一个基本工具，其基本思想是用位姿空间中的一个点来表征机器人的位置和方向。位姿空间法由 W.E.Howden^[17]、S.Udupa^[18]、T.Lozano_Perez 和 M.A.Wesley^[19]、R.A.Books^[20] 等人发展的一种无碰路径规划算法。其实质是根据运动物体的大小和姿态，把周围的障碍物向外扩展一定的距离，即相应的“膨胀”，变成扩展障碍。与

此同时, 运动物体缩小为一个点(运动物体位姿的描述简化为位姿空间的一个点), 于是得到了一个新的空间, 成为位姿空间。由于环境中障碍物的存在, 运动物体在位姿中相应的有一个障碍区。这样把原来在物理空间中求运动物体的无碰撞问题, 变换成位姿空间中求一个质点的运动路径。这实际上是构造了一个虚拟的数据结构, 把运动物体、障碍物及几何约束关系作了等效变换, 简化了问题的求解。位姿空间法把原来求一般物体无碰撞路径问题, 转化为求质点在位姿空间的安全问题, 使问题得到简化, 因此得到了广泛的应用。

目前, 最常用的方法是可视图法^[21], 即在一个无向图中, 将机器人的起始点和终止点以及轨迹中的各点表征为点的形式, 从而机器人的有效路径就是这些点与障碍物不相交的相互连接的线段。如果机器人的尺寸大小忽略不计, 很有可能使得机器人通过障碍物顶点时离障碍物太近甚至接触, 而且该算法在障碍物比较密集的环境搜索时间长。因此, 可视图法虽然可以求得最短路径, 但缺乏灵活性, 且存在组合爆炸问题。后有 Voronoi Diagrams 法和 Tangent Graph^[22]法对可视图法进行了改进。

几何法在二维工作空间下的性能很好, 但扩展到三维工作空间后, 其复杂性和计算时间都会大为增加。

2.2.2 图搜索法

图搜索法采用预先定义的基本形状构造自由空间, 把自由空间切分为一个个的单元。若全部包含障碍物, 称为障碍物单元; 若不包含障碍物则称为自由单元; 若部分包含障碍物, 则称为混合单元。该算法将这些基本单元及它们之间的联系组成一个连通图, 然后运用一定的搜索方法进行路径的搜索。其中主要方法有: 自由空间法和栅格法^[23]。

自由空间法采用预先定义的如凸区法^[24]、三角形法^[25]、广义锥法^[26]等基本形状构造自由空间, 并将自由空间表示为连通图, 通过搜索连通图来进行路径规划。该方法比较灵活, 起始点和目标点的改变不会造成连通图的重构, 但算法的复杂程度与障碍物的多少成正比, 且不是任何情况下都能获得最短路径。为了提高表示的效率, 人们采用层次结构表示。自由空间法的另外一

个缺点就是只在二维空间下性能好，很难扩展到三维的工作环境，一旦扩展到三维的工作环境，其复杂度和计算时间都会大为增加。

栅格法将机器人工作环境分成一系列具有二值信息的网格单元，多采用四叉树或八叉树表示工作环境，并通过优化算法完成路径搜索。该法以栅格为单位记录环境信息，环境被量化成具有一定分辨率的栅格，栅格的大小直接影响着环境信息存储量的大小和规划时间的长短。栅格划分大了，环境信息存储量小，规划时间短，但分辨率下降，在密集环境下发现路径的能力减弱；反之，环境的分辨率高，在密集环境下发现路径的能力强，但环境信息、存储量大，规划时间长。所以该算法的缺点是表示效率不高，存在着时空开销与求解精度之间的矛盾。

后来为了提高表示的效率，人们采用层次结构表示。A.Elfs^[27]提出的网络模型将确定性的网络用于离线的全局规划。Faverjon^[28]采用八叉树表示三维环境的位姿空间，用于机械手的运动规划。Subbarao Kambhampati 采用不完全四叉树表示工作空间，将叶节点连接成一张拓扑图，规划出满足性能的指标路径。路径的搜索策略主要有：A*算法^[29]、D*最优算法^[30]等。

从严格意义上看，位姿空间法也属于图搜索方法的一种。

2.2.3 人工势场法

人工势场法最初是由 Khatib 提出的一种虚拟力法。其基本思想是将机器人在环境中的运动视为一种虚拟的人工受力场中的运动，障碍物对机器人产生斥力，目标点对机器人产生引力，引力和斥力合力作为机器人的加速力，来控制机器人的运动方向。该法简单，便于低层的实时控制，在实时避障和平滑的轨迹控制方面得到了广泛的应用，但对存在局部最优解的问题，容易产生死锁现象（Dead Lock），因而可能使机器人在到达目标点之前就停留在局部最优点。

人工势场法由于其算法的简单和易扩展而引起了越来越多的重视，其基本思想是通过寻找路径点的能量函数（势函数）的极小值点而使路径避开障碍物。通常情况下，其势函数是一个关于距离的标量，在寻找路径点时，若考虑障碍的边界形状，难以用解析式计算而不得不用离散方法搜索此函数的

最快下降方向，从而带来了离散精度和计算量的矛盾，若仅把障碍近似为圆形物体，又难以区别不同边界障碍的作用。而且传统的人工势场法由于没有引入优化过程而容易陷入局部极小区域且不易调节，因而不适于寻求最短路径。在考虑到机器人的动力学约束时，机器人如何跟踪规划出的路径也是一个问题。文献^[11]考虑移动机器人的动力学约束，提出机器人路径规划的最优控制方法，从而使得机器人在避开运动障碍的同时，以最优性能指标完成指定任务。但是，该方法在进行最优控制计算时，初始点的选择受到严格限制，算法的计算时间长，难以保证避障规划的实时性。文献^[12]提出移动机器人最优路径规划新算法，解决了最优路径规划的初始点选择问题，大大提高了算法运算速度，实现了机器人在运动障碍环境中的实时路径规划。

2.2.4 模糊逻辑算法

当环境信息完全已知时，移动机器人路径规划问题的研究已涌现了大量有效的方法。近年来，人们更多地重视动态环境下和存在不确定障碍的路径规划问题。在实际运用中，障碍物的移动往往是随机的，不确定环境中作业的移动机器人在执行任务前通常不能得到环境的全部障碍信息，故无法使用传统的基于精确信息的路径规划方法。特别是动态环境下，虽然可以频繁地进行重复规划，但由于全局规划的计算负担很重，导致规划跟不上实际障碍信息的变化。

现在一种比较有效的方法是不寻求建立机器人工作环境的精确模型，而直接将传感器得到的有限环境障碍信息送给机器人。机器人直接由这些信息根据预先建立的一些规则做出反应。这种方法的出发点是模拟一些低等动物的行为，遵循刺激—反应这样简单的行为规律，而不必直接试图让机器人有人那样高的智能。应用表明，用这种方法实现的自主机器人虽然智能程度较低，但具有很大的实用性，工程可实现性也较强^[13]。规则的建立较多地采用了专家系统、模糊控制的方法。

基于环境反应的方法^[14]不需要环境的数学模型来处理各种情况。该方法最大的特点是参考人的驾驶经验，计算量不大，易做到边运动边规划，能够满足实时性要求，克服势场法易产生的局部极点问题，效果比较理想。但该

方法在复杂环境中难构造出比较全面的规则库，且环境变化较大时，需要花费大量时间来调整和修改已构成的规则库，不具备适应能力。

2.2.5 具有自学习能力的算法

具有自学习能力的算法是能够使机器人具有自学习能力，使机器人在运行中通过与环境的交互或者是人的指导，逐步提高对环境的适应能力的方法。

“学习”这一术语指的是机器人通过试验、观察和推测来更新知识的能力。机器人只有通过不断的学习，才能完善自身的适应能力；依靠与环境不断的交互来获得知识。就研究现状来看，比较有前途的几种方法是用人工神经网络、遗传算法、强化学习或者是这几种方法相结合的研究方法。

J.Holland 在上世纪 60 年代初提出了遗传算法，以自然遗传机制和自然选择等生物进化理论为基础，构造了一类随机化搜索算法。它利用选择、交叉和变异等遗传操作来培养控制机构的计算程序，在某种程度上对生物进化过程做数学方式的模拟。它不要求适应度函数是可导或连续的，而只要求适应度函数为正，同时作为并行算法，它的隐并行性适用与全局搜索。多数优化算法都是单点搜索算法，很容易陷入局部最优，而遗传算法却是一种多点搜索算法，因此更有可能搜索到全局最优点。由于遗传算法的整体搜索策略和优化计算不依赖于梯度信息，所以解决了一些其它优化算法无法解决的问题。但遗传算法的运行速度不快，进化众多的规划要占据较大的存储空间和运算时间。

2.3 路径规划方法的新发展

上边介绍的是一些基本的路径规划方法，每种方法都有各自的优点和不足之处及一定的适用范围。目前，智能机器人领域的研究者们又在探讨新的规划方法。

总体来讲，目前路径规划方法向着互相结合，取长补短的前进方向发展，即向着两种或多种已有算法的有机结合的方向发展。现在，随着移动机器人的路径规划环境趋于复杂，对其规划速度的要求越来越高，并且由于要求机器人完成任务（侦察、巡逻、快速航行）的不同，规划时的性能指标要求趋

于多变和复杂,国内外的研究由以前的二维、已知、静态环境中的规划转向三维、部分已知或未知、变化、有多种因素制约的环境中的路径规划,并借鉴和结合各学科(如仿生学、机械学、生命科学、物理学、经济学等)中的优化方法、智能决策等技术来设计路径规划算法。主要的研究进展有以下几个方面:

(1) 将上节所介绍的几种路径规划方法相结合,又产生了几种新的路径规划方法。例如,将栅格法和单元树法相结合用于解决二维和三维空间的路径规划问题^[35];将八叉树法和人工势场法相结合来解决动态环境下三维空间的路径规划问题^[36];以及对人工势场法进行改进等等^[37]。

(2) 遗传算法由于其具有优良的全局寻优能力和隐含的并行计算特性,所以越来越受到重视。将遗传算法与已有的其它的路径规划方法相结合来解决路径规划问题,取二者之所长,提高了路径规划问题的求解质量和求解效率。例如,遗传算法与栅格法相结合^[38],采用栅格法对机器人工作空间进行划分,用序号表示栅格,并以此序号作为机器人路径规划参数编码,用遗传算法对机器人路径规划进行研究;遗传算法与凸区法的结合^[39],先用凸区法进行粗路径的搜索,再用遗传算法进行路径节点的调整,从而规划出机器人的行走路线;以及遗传算法与人工势场法的结合^[40]等。

但是,遗传算法虽然能从概率的意义上以随机的方式寻求到全局最优解,但它在实际应用过程中也可能会产生一些问题。这些问题最主要的是早熟现象、局部寻优能力差等。它们在路径规划中的典型表现是:得到的路径虽然总体上是较好的,但存在着个别不必要的尖峰(遗传算法使得高于群平均的模式在下一代中获得较多的取样,这样不断进行,一旦某些模式取样在群中占有优势,遗传算法就会强化这种优势,从而使搜索范围迅速变窄,表现为群收敛向一些相同的串),在局部上的爬山能力较差。

(3) 神经网络和模糊逻辑在路径规划中的应用也得到了广泛的重视,关于这方面的文章也很多。例如 Kimmo Pulakka 和 Veli Kujanpaa 利用 SOFM 神经网络来进行路径规划的方法^[41]等。

(4) 多数情况下,局部路径规划的目的是要快速避开先前未知的障碍物,在这方面,比较典型和有效的办法是人工势场法及从它衍生的雷达法^[42]等。很多研究人员都把各种含有学习的算法引入局部路径规划。如:模糊理论^[43]、人工神经网络^[41]、动态二叉树法^[44]、遗传算法^[38] [39]、强化学习^[45]等。从某种角度看,蚁群算法^[46]也是一个具有学习特性的算法。但带有学习性质的算法有些收敛速度慢,有些需进行多次训练才能较好地解决路径规划问题。

2.4 本文提出的路径规划方法

本文作者给出了一种包含全局路径规划和局部路径规划的路径规划解决方案。其中提出了一种解决全局路径规划问题的方法——基于分层模型的遗传模拟退火算法路径规划，以及一种解决局部路径规划问题的方法——基于雷达法的局部路径规划。下面就分别简要介绍这两种方法的基本思想。

2.4.1 基于分层模型的遗传模拟退火算法路径规划

八五和九五期间，“军用智能水下机器人”项目中关于 AUV 全局路径规划问题的研究已经取得了丰硕的成果。八五期间提出的“基于区域模型的路径规划算法”^[47]较好地解决了拥挤环境下的智能机器人全局路径规划问题；九五期间提出的“旋转坐标系法”以及“激活值传播算法”^[48]较好地解决了在海洋环境中考虑海流因素时的全局路径规划问题，其正确性已经通过海试而得以验证；十五期间，“军用智能水下机器人”项目在技术目标方面提出了新的要求，要求之一就是智能水下机器人的工作空间将由九五期间的几公里延伸到 50~100 公里，十五前期提出的“基于局部模型的分层路径规划算法”进一步地解决了这一大范围路径规划的问题^[49]。

十五前期提出的分层路径规划方法的基本思想是：

将机器人的位形空间递归地分解为相继层次上的单元，每个单元都可能属于下列三种情况之一：

- (1) 该单元完全在障碍区之外，标记为 ‘W’，称为白区；
- (2) 该单元完全在障碍区之内，标记为 ‘B’，称为黑区；
- (3) 不是以上两种情况，则标记为 ‘G’，称为灰区。

每层规划，在白区和灰区中寻找一条连通初始位形和目标位形的“通路”，如果得到的通路中含有灰区，则将通路中的单元继续分解为更小的单元，标记它们并重复上面的搜索过程，直到找到一条只含白区的通路，或者确定没有通路存在，或者灰区的尺寸小于给定的精度不能继续分解为止。取通路中的单元的中心点，加上机器人在该单元中的姿态，就得到了一条从初始位形到目标位形的路径。

在我们的系统中机器人有四个自由度： X, Y, Z 三个方向的平移自由度和绕 Z 轴的旋转自由度。假定用 $W=R^4$ 来表示机器人的位形空间，*Source*表示机器人的初始位形，*Target*表示机器人的目标位形，分层路径规划算法通过在连续层次上分解 W 并在分解得到的单元所组成的连通图中搜索得到一条连通初始位形和目标位形的通路。假设用 $W_i, i=0, 1, \dots$ ，来表示 W 在连续层次上分解得到的位形空间， $W_0=W$ 。每个 $W_i, i>0$ ，都是通过分解 W_{i-1} 中的若干个单元而其它单元保持不变得到的。在每一层位形空间 W_i 构造好之后，计算 W_i 中的单元之间的连通关系而得到一张连通图 G_i ，通过搜索这张连通图而得到一条连通初始位形*Source*和目标位形*Target*的通路。分层路径规划算法的简单的实现方法如下：

- (1) 初始化，位形空间 $W_0=W$ ，构造该层位形空间上的连通图 $G_i, i=0$ 。
- (2) 在 G_i 搜索一条连通初始位形*Source*和目标位形*Target*的通路，如果找到一条只含白区的通路，则算法结束；如果找到一条含有灰区的通路，则转(3)；否则，算法失败。
- (3) 假定用 L 表示在第(2)步得到的含有灰区的通路，对于 L 中的每一个单元 k ，分解 k 并将分解得到的单元加入 W_i 得到 W_{i+1} ，构造相应于 W_{i+1} 的连通图， $i=i+1$ ，转(2)。

根据上面的算法，我们可以看出由于分层算法通过在每层建立并搜索连通图而得到一条连通初始位形*Source*和目标位形*Target*的通路。随着位形空间向更低的层次的进一步分解，位形空间以及与之相对应的连通图中的节点越来越多，因此保存这些节点所需的内存空间也就越来越多。本课题组的成员将遗传算法应用于机器人路径规划问题的领域进行了尝试与研究^[50]，提出了一种将分层思想与遗传算法相结合的方法，将每层各单元间的连通信息体现在遗传个体的编码中，不用建立连通图，从而在一定程度上节省了内存空间。

但是，遗传算法虽然能从概率的意义上以随机的方式寻求到全局最优解，但它在实际应用过程中也可能产生一些问题。这些问题最主要的是早熟现象、局部寻优能力差等，另外遗传算法强调两代之间的进化关系，其交配可能使最好的解遗失。这些缺点在路径规划中的典型表现是：得到的路径虽然总体上是较好的，但存在着个别不必要的尖峰（遗传算法使得高于群

平均的模式在下一代中获得较多的取样, 这样不断进行, 一旦某些模式取样在群中占有优势, 遗传算法就会强化这种优势, 从而使搜索范围迅速变窄, 表现为群收敛向一些相同的串), 在局部上的爬山能力较差。

模拟退火 (Simulated Annealing 简称 SA) 算法是由 Kirkpatrick 于 1983 年提出的^[51], 它来源于物理退火过程, 该算法利用具有概率突跳性的 Metropolis 抽样准则在解空间进行随机搜索, 伴随温度的不断下降而重复抽样的过程, 最终得到优化问题的全局最优解。它以初始温度足够高, 下降足够慢以及每一温度下抽样次数足够多为前提条件, 因而收敛速度非常慢。应用起来有一定的困难。

总结遗传算法和模拟退火算法的优缺点, 本文作者将两种算法结合并应用于机器人路径规划问题, 提出了一种基于分层模型的遗传模拟退火算法路径规划方法。利用模拟退火算法的突跳性, 可解决遗传算法局部爬山能力差的问题; 另外模拟退火算法的概率接受新的个体的策略保留了个体的多样性, 解决了遗传算法交配时丢失最好解的问题和收敛过快的问题。算法的基本思想和具体实现方法将在下一章详细阐述。

2.4.2 基于雷达法的局部路径规划

在前面我们介绍了单元分解法的基本思想, 其中提到栅格法。该方法的特点是思想简单, 易于实现, 并且易于扩展到三维环境。可它的主要缺点是对工作区域的大小有一定的要求, 如果区域太大, 将使栅格的数量急剧增加, 使搜索存在组合爆炸的问题。因此大范围的全局路径规划问题不适合用栅格法来解决。可是一般的局部路径规划是根据全局规划的结果进行的, 规划范围不会很大, 只是在全局规划出的路径相邻两点间进行, 而且栅格法比起其他的算法而言搜索过程十分方便灵活, 所以采用栅格法来解决局部路径规划问题既简单又实用。同时, 海洋环境下海流情况复杂, 机器人在行进中很容易被海流冲偏航向, 甚至无法到达预定位置, 因此在规划中必须考虑海流的影响。由于全局路径规划是针对已知条件的先验性算法, 不适合解决海洋中的实时海流问题, 我们考虑在局部路径规划算法中解决机器人行进过程中的海流问题。

雷达法正是基于栅格模型的一种路径规划方法，它是受启发于雷达的工作原理，具有在未知、部分已知和变化的环境中进行路径规划的能力。雷达法路径规划是基于这样一种基本思想：雷达波在各个方向的传播速度都是一样的，并且假设该雷达波的波长无限长，则当遇到障碍物时，将发生类似波形的干涉现象。在本文中，我们把机器人的活动区域按照栅格法进行环境的建模，即工作环境被表示成大小相等的 2^N 个栅格。雷达法来自这样一个简单思想：在 Ψ 中给定起终点 S , T ，终点 T 等时间间隔地发出等速的波沿在 Ψ 中传播，波沿不能穿过障碍物。 S 最先收到波沿的邻点 P 一定是到 T 的理想路径中的一部分，这是因为波沿是朝各个方向等速传播的，所以如果 P 是第一个收到波沿的，它就在一个比其它后收到波沿的点所在路径短的路径上。于是 P 被加入到 Ω 中，并成为新的起点，如果现在 Q 是 P 最先收到波沿的邻点，则 Q 也加入到 Ω 中，并成为新的起点，如此下去，每一个波沿产生一个路径点，直到到达 T ，一条从 S 到 T 的优化路径就形成了。在局部路径规划方面，该算法可以说是人工势场法的一个衍生算法，它的特点和人工势场法一样，原理简单，容易实现。考虑到局部路径规划要求对随机出现的障碍物和海流等复杂情况进行及时反应和处理，该算法利用声纳传感器实时得到的障碍物和海流等环境信息动态调整栅格模型，在机器人行进过程中根据周围栅格的权值快速加以判断并决定下一步移动的方案。算法的基本思想和具体实现方法将在第四章详细阐述。

2.5 本章小结

路径规划是智能机器人领域核心技术之一。本章就机器人路径规划技术展开了讨论。首先简要的介绍了路径规划问题的描述、分类及特点、相关问题及一些的路径规划方法，然后在此基础上提出了本文所讨论的全局路径规划和局部路径规划的方法。

十五期间对机器人提出的新的要求是机器人的工作空间要延伸到 50~100 公里，因为工作空间的扩大，必须使用一种能够解决大范围环境下的路径规划问题的算法。同时，海洋环境下海流情况复杂，机器人在行进中很容易被海流冲偏航向，甚至无法到达预定位置，因此在规划中必须考虑海流

的影响。根据这些特殊性，本文作者提出了一套包括全局和局部路径规划的算法。作者以分层模型为基础，将遗传模拟退火算法应用于大范围路径规划问题，设计出了全局路径规划算法。该算法适用于解决大范围路径规划问题，并考虑了机器人行进中的能量消耗。由于全局路径规划是针对已知条件的先验性算法，不适合解决海洋中复杂的实时问题，因此我们提出了适用于解决海流、避障等问题的基于雷达法的局部路径规划算法。基于分层模型的遗传模拟退火算法路径规划方法和基于雷达法的局部路径规划方法将在第三章和第四章中分别进行详细介绍。

第3章 基于分层模型的全局路径规划

全局路径规划是智能水下机器人自主导航中的关键环节之一，在一定程度上它影响着水下机器人指定作业的完成。水下机器人全局路径规划的任务就是给定一个水域及障碍环境、一个运动的水下机器人及其起始位形和目标位形，求一条从起点到目标的路径，使该水下机器人能够从起点安全无碰地绕过障碍物到达终点。全局路径规划需要知道关于环境的所有消息，根据环境地图进行大粒度的路径规划，并产生一系列关键点作为子目标点下达给局部路径规划系统。

前述的多种路径规划方法大多数只适用于二维空间，三维空间特别是大范围环境中的路径规划十分复杂。而智能水下机器人所处的环境就属于大范围的开放的三维空间，十五期间对机器人提出的新的技术目标之一就是要求机器人的工作空间能够延伸到 50~100 公里。同时，考虑到客观条件的限制和实际应用的需要，机器人运动姿态与能量消耗、求解精度与内存空间之间的矛盾等等也都是全局路径规划中要考虑的因素。遗传算法是一种传统的解决最优化问题的方法，目前已经有很多用于解决机器人二维路径规划问题的成功例子，如文献[52]。模拟退火算法的突跳性可解决遗传算法局部爬山能力差的问题；另外模拟退火算法的概率接受新的个体的策略保留了个体的多样性，解决了遗传算法交配时丢失最好解的问题和收敛过快的问题。但是由于遗传模拟退火算法收敛较慢，在大范围的路径规划中并不能很好的发挥作用，并且难于扩展到三维空间。基于分层模型的遗传算法路径规划方法较好地解决了上述问题，下面就介绍一下算法的基本思想及算法实现中的几个关键技术。

3.1 算法概述

在我们的“XX 智能水下机器人”课题路径规划的研究中，机器人不可能被当作一个质点来考虑，这是由环境空间的约束和机器人本身形态因素所决

定的。因此，机器人的位置和姿态都是路径规划中需要考虑的因素，即在多个自由度的条件下为机器人进行路径规划。

描述一个机器人的位姿需要标识出其所有的自由度，包括平移和旋转。在 k -维空间中，一个机器人的位姿可以用几个相互独立的参数来描述，这些参数完全可以表示机器人上任意一点的位置。我们选择机器人上一个用全局坐标来表示的点作为参考点。这时，描述机器人的位姿需要 $d = k + \binom{k}{2}$ 个参

数。其中， k 个参数用来表示 R^k 中参考点的坐标， $\binom{k}{2}$ 用来标识机器人的转向。例如， R^2 中一个机器人的位姿要用 (x, y, θ) 来表示， $d = 2 + 1$ ， (x, y) 表示参考点坐标， θ 表示机器人的旋转自由度。如果把机器人当作一个点来考虑，即忽略 θ ，则 $d = 2$ ，机器人只有两个方向的自由度。

按照上面的计算方法，在三维海域中，机器人本应有六个自由度： X, Y, Z 三个方向的平移自由度和分别围绕 X, Y, Z 三个轴旋转的旋转自由度。就本文所研究的水下智能机器人来说，其试验平台安装有六个推进器，可做出上下，左右，前后及简单的组合运动。为使实际控制简单，易于实现，我们只考虑机器人在 X, Y 和 Z 方向的平移自由度及其绕 Z 轴旋转（即在 XOY 平面上改变艏向）的旋转自由度。这时，机器人的位姿可以用一个四元组 (x, y, z, θ) 来描述，其中 (x, y, z) 表示机器人的参考点在三维工作空间中的位置坐标， θ 表示机器人的姿态，即机器人绕 Z 轴所旋转的角度，用机器人艏向与 X 轴正向的夹角的弧度值来表示。最后算法生成的路径也将用该四元组来表示。

我们把机器人从当前位置以某种运动形式运动到下一位置看作是一种能量消耗的过程。从节能的角度考虑，算法需要根据机器人行进中的能量消耗进行规划。

3.1.1 区域模型

3.1.1.1 环境表示方法

机器人在规划前首先要做的就是将环境的描述由外部的原始形式通过一系列处理转化为适合规划的内部的世界模型，这个过程我们称为环境建模，

定的。因此，机器人的位置和姿态都是路径规划中需要考虑的因素，即在多个自由度的条件下为机器人进行路径规划。

描述一个机器人的位姿需要标识出其所有的自由度，包括平移和旋转。在 k -维空间中，一个机器人的位姿可以用几个相互独立的参数来描述，这些参数完全可以表示机器人上任意一点的位置。我们选择机器人上一个用全局坐标来表示的点作为参考点。这时，描述机器人的位姿需要 $d = k + \binom{k}{2}$ 个参

数。其中， k 个参数用来表示 R^k 中参考点的坐标， $\binom{k}{2}$ 用来标识机器人的转向。例如， R^2 中一个机器人的位姿要用 (x, y, θ) 来表示， $d = 2 + 1$ ， (x, y) 表示参考点坐标， θ 表示机器人的旋转自由度。如果把机器人当作一个点来考虑，即忽略 θ ，则 $d = 2$ ，机器人只有两个方向的自由度。

按照上面的计算方法，在三维海域中，机器人本应有六个自由度： X, Y, Z 三个方向的平移自由度和分别围绕 X, Y, Z 三个轴旋转的旋转自由度。就本文所研究的水下智能机器人来说，其试验平台安装有六个推进器，可做出上下，左右，前后及简单的组合运动。为使实际控制简单，易于实现，我们只考虑机器人在 X, Y 和 Z 方向的平移自由度及其绕 Z 轴旋转（即在 XOY 平面上改变艏向）的旋转自由度。这时，机器人的位姿可以用一个四元组 (x, y, z, θ) 来描述，其中 (x, y, z) 表示机器人的参考点在三维工作空间中的位置坐标， θ 表示机器人的姿态，即机器人绕 Z 轴所旋转的角度，用机器人艏向与 X 轴正向的夹角的弧度值来表示。最后算法生成的路径也将用该四元组来表示。

我们把机器人从当前位置以某种运动形式运动到下一位置看作是一种能量消耗的过程。从节能的角度考虑，算法需要根据机器人行进中的能量消耗进行规划。

3.1.1 区域模型

3.1.1.1 环境表示方法

机器人在规划前首先要做的就是将环境的描述由外部的原始形式通过一系列处理转化为适合规划的内部的世界模型，这个过程我们称为环境建模，

其中主要是障碍物的表示方法。合理的环境表示才能有利于规划中搜索量的减少,才能有利于时空开销的减少。不同的规划方法正是基于这种不同的环境建模。

图 3.1 给出了几种常用的环境表示方法^[53],其中(a)是原始的工作空间。

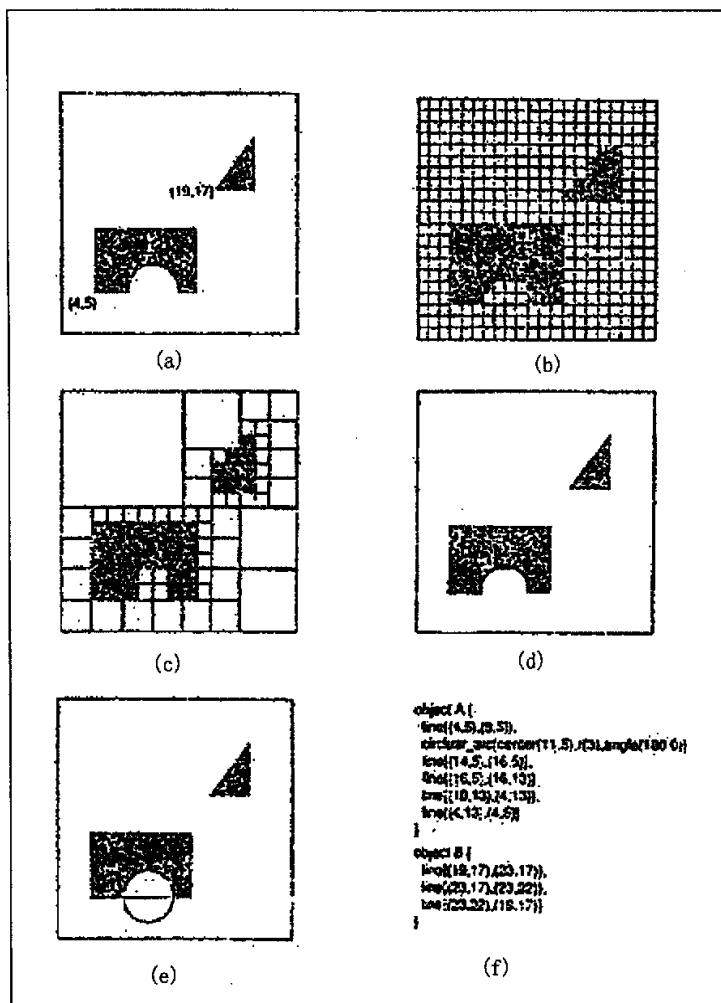


图 3.1 环境的表示方法

图 3.1 中的(b)是用栅格法来表示环境的。使用大小相同的栅格划分机器人的工作空间,并用栅格数组来表示环境。每个栅格点标以两种状态之一,或者在自由空间中,或者在障碍空间中。如图中所示,黑格代表障碍物,在

栅格数组中标为 1；白格代表自由空间，标为 0。最短路径是通过搜索这张栅格图得到的。为了提高搜索的效率，栅格通常按粒度分成若干层次。这种方法的特点是简单，易于实现，从而为路径规划器的实现带来了很大方便，具有表示不规则障碍物的能力，适合于大规模并行处理机的实现。其缺点是表示效率不高，存在着时空开销与求解精度之间的矛盾。

单元树法正是为了克服栅格法的缺点而设计的。这种方法把机器人工作空间划分成几个较大的单元（一般来说，二维空间划分成 4 部分，称为四叉数；三维空间划分成 8 部分，称为八叉树），如图 3.1 中(c)。划分得到的每个单元所占用的工作空间可能是下面三种情况之一：都为自由空间；都为障碍物空间；混合型空间，即既包含了障碍物区域，又包含了自由区域。对于最后一种类型的单元按照前面的方法继续进行划分，直到一个预先设定好的精度为止。该方法的主要缺点是计算单元之间的邻接关系时的损失较大。

图 3.1 中(d)所表示的多边形表示法，这也是常用的方法之一。该方法用多边形来逼近障碍物，并使用了很多成熟的诸如求交叉点和测距等方面的解析几何算法。

图 3.1 中(e)所示的是 CSG 法，同 CAD 表示部件的方法相似。(f)所示的 B-rep 是用障碍物的边缘函数来表示环境的方法。应用这两种环境表示方法的路径规划算法较少。

3.1.1.2 区域模型的构造

分层的思想最初由 Brooks 和 Lozano-Perez 在文献[54]中提出来的，以后又有一些作者对它进行了改进^{[55][56][57][58][59]}。文献[60]在单元分解思想的基础上结合 Brooks 和 Lozano-Perez 的分层的思想，设计出了一种分层路径规划方法。为了解决大范围规划的问题，本算法区域模型的构造基于分层思想，下面简要介绍一下。

在划分区域时，如果环境中障碍物形体复杂，则划分出的区域非常细碎，会占用大量的存贮空间，搜索过程也会很慢。因此我们在构造区域模型时，先为障碍物外接一个直六面体，再进行区域划分。这样不仅大大减少了所占用的空间和时间，简化了区域的划分，而且对保证机器人运动的安全性也是有好处的。

将一个大的区域分解为小的区域的集合，方法有很多种，这里我们采用一种简单的方法，即将其分解为 2^m 个大小相等的区域，这里 m 是机器人工作空间的维数，通常二维空间分解为四个区域，三维空间分解为八个区域。采用这种区域划分的方法要求机器人的工作空间是由 $2^N \times 2^N \times 2^N$ 个基本元素组成的正方体区域，如果不满足，可以对原工作空间进行拓展，并将拓展部分设置为障碍区。每个基本元素的形状也是正方体，并且其尺寸不小于机器人长、宽和高的最大值。

对于每个分解得到的区域，根据其其与障碍物的位置关系，可以分为黑区、白区和灰区三种类型。一个区域如果完全在障碍区之内，则称为黑区，标记为‘B’；一个区域如果完全在障碍区之外，则称为白区，标记为‘W’；否则，称为灰区，标记为‘G’。图 3.2 给出了一个区域分解的实例。

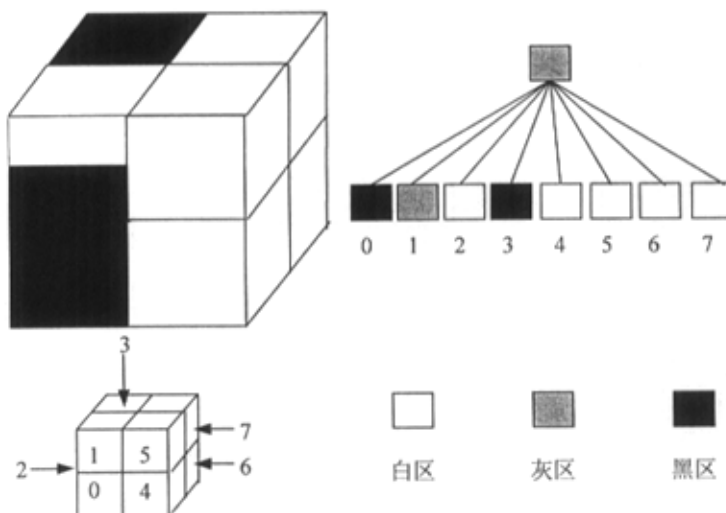


图 3.2 区域划分实例

区域划分完毕后，为了使这种区域模型利于机器人的路径规划，对每一个区域存贮以下属性值：

- (1) 区域的编号 num 。
- (2) 区域的长度 $length$ 。
- (3) 区域中心点的坐标 (x, y, z) ，最后生成的路径就是由得到的只含有白

区的通路中的各区域的中心点的坐标 (x, y, z) 以及机器人在该区域中的姿态 θ 组成。

- (4) 指向该区域的父区域的指针。
- (5) 指向该区域的 8 个子区域的指针。
- (6) 区域的属性：白区、灰区或黑区。
- (7) 区域划分标记，该区域已经划分过，则为 *True*，没有划分过，则为 *False*。
- (8) 失败标记 *failure*。用于失败处理。关于失败处理，将在下面详细介绍。

从所保存的区域模型的信息中我们可以看出，没有体现区域之间的连通性和机器人的姿态信息，只保存了区域本身的一些属性，区域之间的连通信息和机器人的姿态将在搜索路径时考虑，这样就大大减少了对内存空间的利用。

本章所介绍的路径规划算法用分层单元分解法所表示的三维空间作为基础环境模型，应用遗传模拟退火算法的搜索方法来搜索路径。分层路径规划算法的基本思想是将机器人的工作空间按上节描述的方法递归地分解为相继层次上的区域，逐层运用一定的路径搜索算法在白区和灰区中寻找一条连通初始位形和目标位形的路径，如果得到的路径中含有灰区，则继续分解该区域为更小的区域，标记它们并寻找路径，直到找到一条只含白区的路径，或者确定没有路径存在，或者灰区的尺寸小于给定的精度。

分层单元分解算法如下：

- (1) 初始化。机器人的整个工作空间构成第一层区域，分解这个区域，确定子起点和子终点，然后在连通图中搜索第一层通路 L_1 ， $i=1$ 。
- (2) 若通路 L_i 中的每个节点的区域类型都为白区，则返回通路，转(5)，否则，转(3)。
- (3) 对第 i 层通路 L_i 中的每个节点的区域 j 做如下处理：分解区域 j ，确定子起点和子终点，然后在分解完的区域中按照一定搜索算法搜索一条子通路 π_j ， π_j 由区域 j 中类型为白区或灰区的节点组成，并且满足以下条件：如果区域 j 为通路 L_i 的第一个节点的区域，则 π_j 的第一个节点的区域包含起点；如果区域 j 为通路 L_i 的最后一个节点

的区域, 则 π_j 的最后一个节点的区域包含终点。将得到的子通路 π_j 加入第 $i+1$ 层通路 L_{i+1} 。

(4) $i=i+1$, 转(2)。

取出通路中的每个节点的区域中心点坐标 (x, y, z) 和节点的姿态 θ , 组成一个四元组 (x, y, z, θ) , 这些四元组就构成了一条从起点到终点的安全路径。

由分层规划算法可知, 我们将机器人的工作空间递归地分解为若干层次, 在同层的各个区域中, 分别运用路径搜索算法在白区和灰区中寻找一条连通初始位形和目标位形的子路径, 各个区域的子路径合起来组成该层的路径, 重复分层和搜索, 直至找到一条连接起点和终点的安全路径或发现无法找到路径则算法结束。因此一个大范围的搜索就变成了在一个个分解后的小区域中的搜索。

从上面的描述我们可以看到分层规划必须结合一定的搜索算法在每层中搜索最优路径。文献[60]在每层中构造连通图, 根据连通图使用激活值传播算法搜索子路径。每次连通图的构造和存放既耗费时间, 又占用内存。文献[50]中采用遗传算法搜索子路径, 将每层各单元间的连通信息体现在遗传个体的编码中, 不用建立连通图, 从而在一定程度上节省了内存空间。但是由于遗传算法的局限, 所得到的路径虽然总体上是较好的, 但存在着个别不必要的尖峰, 局部上并非最优。本文采用遗传模拟退火算法, 模拟退火算法具有摆脱局部最优点的能力, 利用这一点, 可以有效地解决上述问题。下面就介绍一下搜索算法的设计和实现。

3.1.2 搜索算法—遗传模拟退火算法的实现

3.1.2.1 遗传算法在路径规划中的应用

利用遗传算法进行机器人路径规划时, 须解决的很多关键性问题, 如遗传算法都包含的基本操作、各种遗传参数的设置等等。要较好地进行遗传算法的寻优过程, 就要对这些问题进行深入地研究并进行试验, 设计并确定问题的解决方案。下面我们就分别介绍在本文算法中要解决的涉及遗传算法的关键问题及方法:

- (1) 规划环境的表示：用分层单元分解法来表示机器人的移动空间。如上节中所述。
- (2) 路径编码：按照区域模型将算法的工作区域划分为八个子区域节点，并将子区域编号。个体表示机器人在区域中的运动路径。将运动路径中的子区域的编号按机器人沿路径行走的先后顺序串起来即为该个体的编码，例如，017 就代表一条由 0 号区域经 1 号区域到达 7 号区域的运动路径，同理，0457 也代表一条连接 0 号区域和 7 号区域的路径，如图 3.3 所示。显然路径中包含的子区域越多，个体编码的长度就越长，因此个体编码的长度是不固定的。但是在路径中没有重复节点的情况下，个体编码的最大长度也不会超过 8 位。

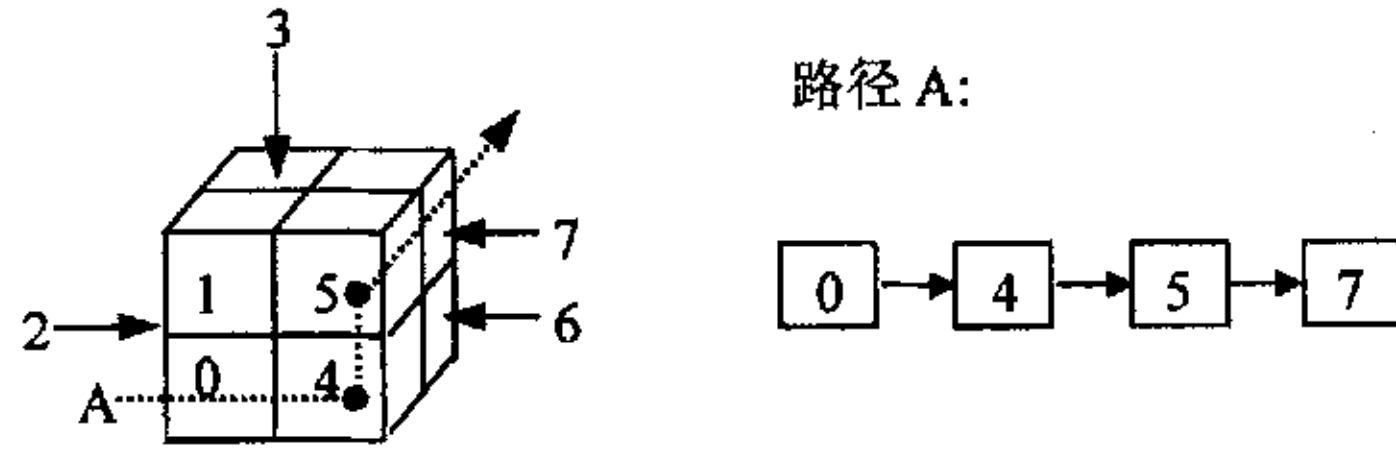


图 3.3 个体编码实例

(3) 适应度函数：

个体评价函数直接影响遗传算法的计算效率和计算时间。根据机器人的实际情况，在设计评价函数时必须兼顾三方面的因素：路径长度、运动消耗与机器人艏向顶流等多方面因素。运动消耗指的是机器人从当前位置以某种运动形式运动到下一位置都消耗了一定能量，每一种运动形式都对应一个能量消耗值。路径长度是将路径中相邻两节点之间的欧氏距离迭加得出整条路径的长度。而艏向顶流是指根据海洋环境下 AUV 路径规划的特殊性，要求机器人尽量顶着海流的方向前进，从而不会被海流冲偏航向。

本算法的评价函数 f 为：

$$f = \begin{cases} P_{\max} \times O_{\max} \times T_{\max} - \sum_{i,j} O_{ij} \times T_{ij} & , \text{当个体有效} \\ 0 & , \text{当个体无效} \end{cases} \quad (3-1)$$

其中： P_{\max} 为路径中最大节点数，通常为 8， O_{\max}, T_{\max} 分别是个体中相

邻节点间的欧氏距离和能量消耗的最大权值； i, j 为路径中相邻节点， O_{ij} 为 i, j 两节点间的欧氏距离， T_{ij} 为机器人由节点 i 运动到节点 j 的能量消耗权值。对于适应度函数的设计在下一节中将详细介绍。

- (4) 选择算子：将种群中的个体进行适应值评价后，运用轮盘赌算法，产生中间代，使个体按一定概率向下一代繁殖。
- (5) 交叉算子：交叉操作包括传统意义上的交叉和重合点交叉。传统意义上的交叉显然易产生间断点。重合点上的交叉指的是对在种群中随机选取的两个有效个体，在两个个体节点编号完全相同的位置进行交叉操作，当重合点多于一个时，随机选择其一进行交叉，当无重合点时，不进行操作。由于是在相同的位置交换有效个体的某一段，交叉后的个体仍是有效的，所以此操作不易产生间断路径。
- (6) 变异算子：三种变异操作。个体中随机删除一个路径点；在个体中随机选择一个节点，用另一个随机产生的点代替；在个体中随机选择一个节点前插入一个新节点。

显然以上算子的操作可能产生路径不连续的情况，也可能产生路径中使路径产生冗余节点；对于第一种不连续的情况，可进行插入中间点的操作使之连续。对于第二种情况，为了节约时空资源，要把冗余节点删除。所以引入以下两种算子：

- (7) 插入算子：将间断路径用自由节点弥补，使之成为连续路径。例如，节点0和节点5不可达，可以插入节点1或节点4，使两节点可达。当可以插入的节点不是自由节点，则插入失败，对应个体为无效个体。插入算子可以使很多无效个体变成有效个体，防止了种群因为有效个体太少而影响进化的正常进行。
- (8) 删除算子：在此引入删除算子。其目的是删除路径中冗余节点，将同一个体中两相同节点之间的节点以及两相同节点中的一个一并舍去。删除算子的引入节约了时空资源，提高了算法的效率。

3.1.2.2 遗传模拟退火算法在路径规划中的应用

正如我们在前面叙述的，遗传算法是一个有效的寻优算法，但是在应用过程中也有自身的缺陷，这些问题中最主要的就是局部寻优差和早熟现象。

遗传算法是以随机的方式寻求全局最优解的, 所以从概率上讲, 遗传算法能够找到全局最优解, 但是往往在算法执行过程中所求得的知识局部最优解, 全局并非最优。引起这些问题的主要原因是, 新一代群体的产生主要是依靠上一代群体之间的随机交叉重组来进行的。所以即使是在最优解附近, 要达到这个最优解, 却要花费较大的代价。而另一方面, 模拟退火算法却具有摆脱局部最优点的能力, 利用其突跳性, 可解决遗传算法局部爬山能力差的问题。另外模拟退火算法的概率接受新的个体的策略保留了个体的多样性, 解决了遗传算法交配时丢失最好解的问题和早熟的问题。所以使用遗传算法与模拟退火算法相结合的方法, 是解决上述问题的有效途径。图 3.4 为基于遗传模拟退火算法的机器人路径规划问题求解算法流程示意图。

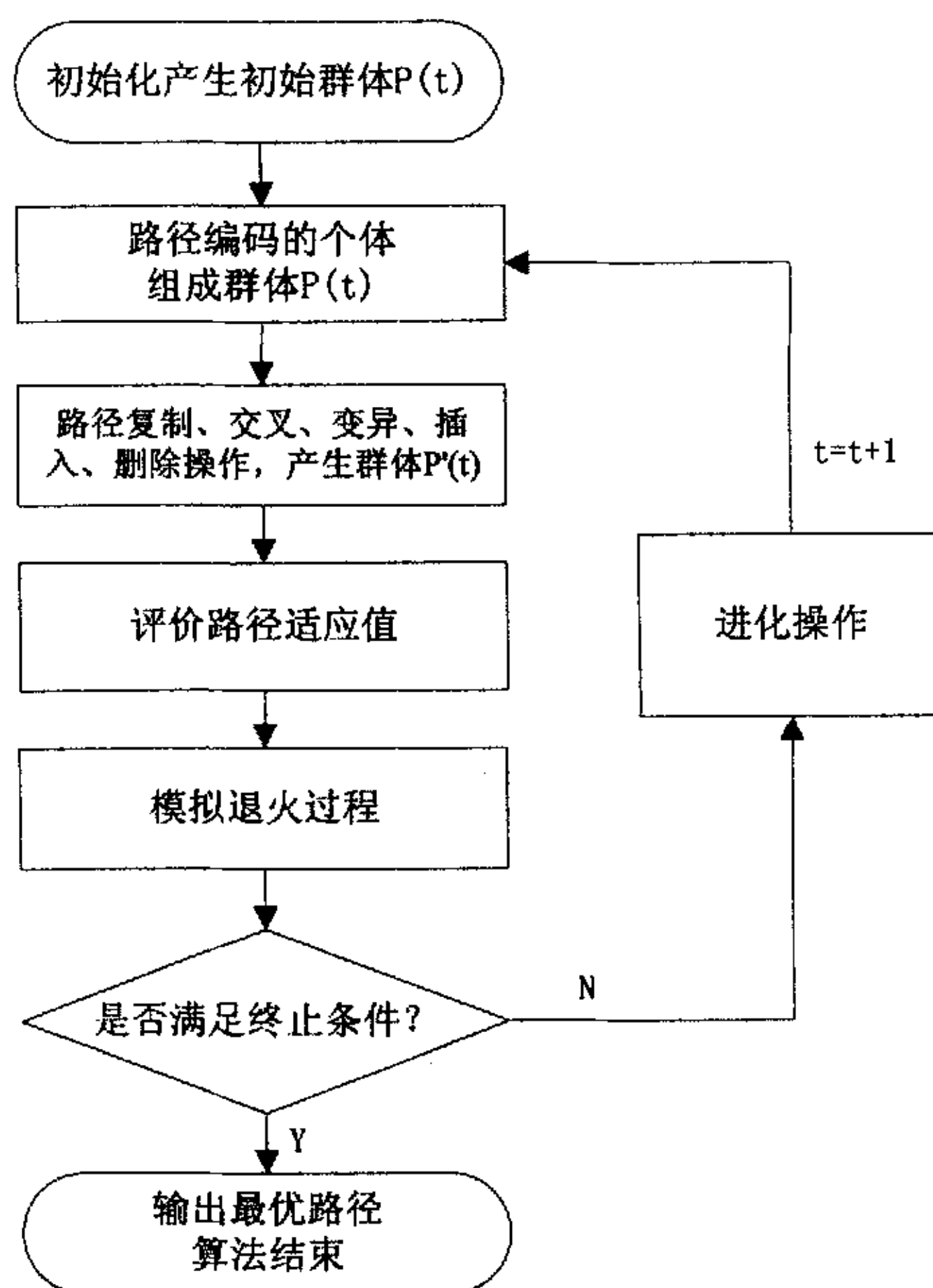


图 3.4 基于遗传模拟退火算法的机器人路径规划
问题求解算法流程示意图

该过程可用下述算法描述:

算法 PlanningPath-GSA

1 初始化。

①确定种群规模、交叉率、变异率等有关遗传参数;根据种群规模随机挑选有效个体组成初始化群体 $P(g)$ 。遗传代数计数器初始化: $t \leftarrow 0$ 。

②设定初始温度,降温表等有关模拟退火参数。设置初始温度参数:
 $T \leftarrow T_{\max}$ 。

2 评价种群中各个个体的适应值: $\{f_{i1}, f_{i2}, \dots, f_{im}\} \leftarrow \text{fitness}[P(t)]$ 。

3 由选择算子从父代个体中选择合适的个体进行子代个体的复制,由交叉算子和变异算子进行个体的交叉和变异,由插入算子保证路径的连续,由删除算子去掉路径中的冗余部分:

$$P_1(t) \leftarrow \text{selection}[P(t)], P_2(t) \leftarrow \text{crossover}[P_1(t)], P_3(t) \leftarrow \text{mutation}[P_2(t)], \\ P_4(t) \leftarrow \text{insertion}[P_3(t)], P'(t) \leftarrow \text{deletion}[P_4(t)].$$

4 评价由 3 得出的各个个体的适应值: $\{f_{j1}, f_{j2}, \dots, f_{jm}\} \leftarrow \text{fitness}[P'(t)]$ 。

5 进行模拟退火过程,可按以下方式进行筛选:如果父代路径中的路径 is 和 it ($s, t=1, 2, \dots, m$) 生成子代路径中的路径 js 和 jt , 则分别以概率 P_s 和 P_t 接受 is 和 it 为新一代路径, 分别以概率 $1-P_s$ 和 $1-P_t$ 接受 js 和 jt 为新一代路径,

由此形成经过一代遗传后的新路径集合。其中: $P_s = \frac{1}{1 + \exp(-\frac{f_{is} - f_{js}}{T})}$,

$$P_t = \frac{1}{1 + \exp(-\frac{f_{it} - f_{jt}}{T})}。$$

6 判定终止条件。判断当前状态是否满足模拟退火停止循环的条件。如果不满足终止条件,则进化代数加 1, 即 $t = t + 1$, 并且按降温方法更新温度参数 T , 转 2; 如果满足终止条件,则输出当前最优路径, 算法结束。

3.2 算法实现中的关键技术

在上一节中介绍了基于分层模型的全局路径规划的基本思想及一般过

程,在这一节中将对算法实现中的几个关键技术进行讨论。

3.2.1 初始种群的设定

由于工作区域只包含 8 个节点,根据编码规则,在初始化种群时,将个体长度固定为 8 位。为了简化计算、缩小搜索范围,初始化种群中的个体均为连接子起点和子终点的随机路径,路径的中间节点均为自由节点。所谓自由节点是指属性为白区或者灰区、非子起点和子终点的节点。鉴于子起点和子终点可直接相连的情况也有可能出现,为了加快算法的进化速度,将只包含起点和终点两个节点的路径加入初始种群。

种群规模影响到遗传算法的最终性能和效率。考虑到算法的特殊性,由于每个区域可选自由节点不多(不超过 6 个),个体编码长度不大。如果固定种群的规模,一方面不好确定合适的尺度,一方面造成内存资源的不必要浪费,因此本算法根据搜索区域中自由节点的个数确定初始种群的规模。

3.2.2 适应度函数的设计

个体适应度函数直接影响遗传算法的计算效率和计算时间。根据机器人的实际情况,在设计评价函数时必须兼顾三方面的因素:路径长度、运动消耗与机器人艏向顶流等多方面因素。运动消耗指的是机器人运动消耗的能量。路径长度是将路径中相邻两节点之间的欧氏距离迭加得出整条路径的长度。而艏向顶流是指根据海洋环境下 AUV 路径规划的特殊性,要求机器人尽量顶着海流的方向前进,从而不会被海流冲偏航向。在全局路径规划中,我们暂时不考虑机器人艏向顶流的因素,这个问题,我们在局部路径规划中进行考虑和解决。

在对个体进行评价之前,先判断它的有效性。若个体有效则根据设计好的评价函数计算适应度,然后取适应度高的个体进行下一轮进化。这样能够简化计算,提高算法的效率。具体方法是:当路径中的相邻节点存在不可达情况,则代表该路径的个体无效,无权进行适应度计算,适应度为零。相邻结点的可达与不可达由机器人的运动形式决定。

我们在这里讨论的机器人有 12 种运动形式:机器人相对其自身可以向前

运动 (Forward), 向后运动 (Backward), 向左运动 (Leftward), 向右运动 (Rightward), 向上运动 (Upward), 向下运动 (Downward) 以及向逆时针方向 (Counterclockwise) 旋转和向顺时针方向 (Clockwise) 旋转, 同时还有简单的合成运动: 左前运动 (Left-Forward), 右前运动 (Right-Forward), 左后运动 (Left-Backward) 和右后运动 (Right-Backward)。图 3.5 所示的是机器人在 XOY 平面上以及逆时针和顺时针旋转之后的邻接情况, 再加上机器人向上和向下运动之后得到的两个相邻节点, 共 12 个相邻节点。其中填充为黑色的是机器人的当前位姿, 白色的为机器人运动之后的位姿。需要注意的是, 机器人只能以一种方式从节点 i 运动到节点 j 。

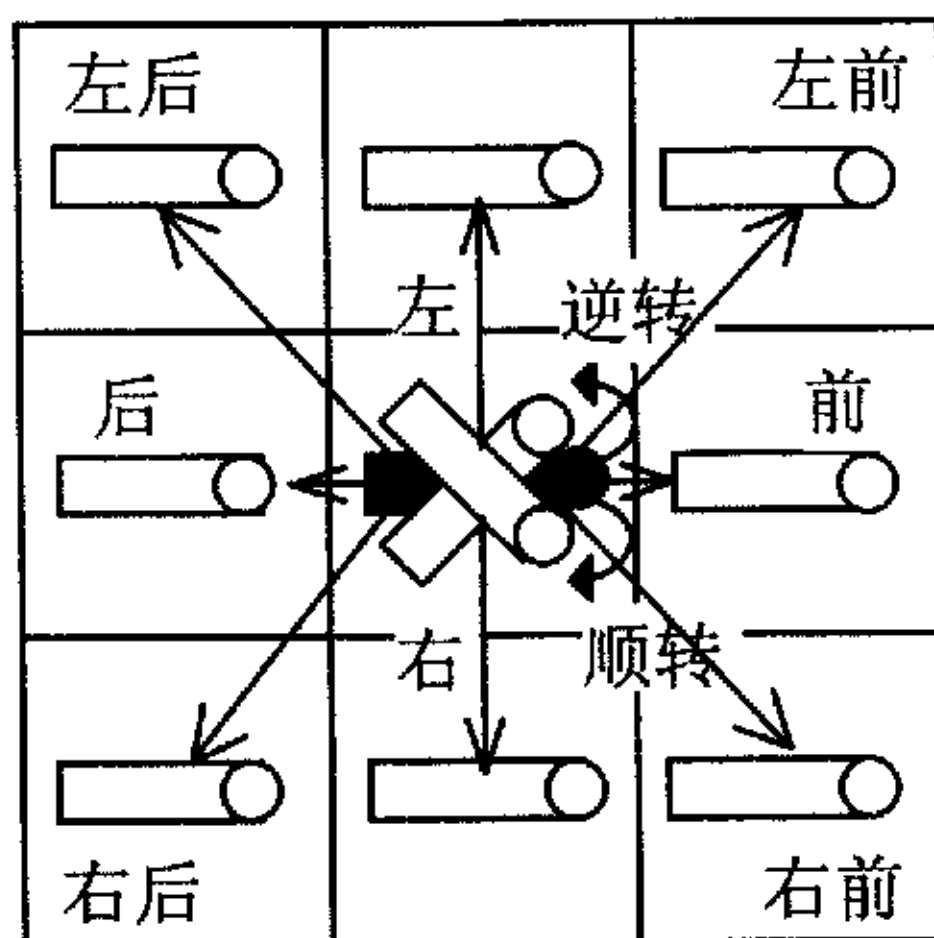


图 3.5 机器人在运动空间中的连通情况

由机器人运动方式的限制可知, 机器人无法作左上、左下、左前上、左前下、左后上、左后下、右上、右下、右前上、右前下、右后上、右后下、前上、前下、后上、后下等复杂的复合运动。则存在多种两个节点不可达的情况, 例如 0 和 5、1 和 4 等等。

运用适应度函数对个体进行评价, 直接影响遗传算法的计算效率和计算时间。根据机器人的实际情况, 在设计适应度函数时必须兼顾路径长度与运动消耗两方面因素。

(1) 运动消耗

机器人从当前位置以某种运动形式运动到下一位置是一种能量消耗的过程。给上节中提到的机器人的每一种运动方式都赋予一权值 T_{ij} , 代表的是机器人在当前节点选择下一步的策略, 因此权值的大小表示机器人作相应运

动的耗能。我们用 12 个浮点型变量来分别表示这 12 种策略：*ForStrength*, *BackStrength*, *LeftStrength*, *RightStrength*, *Left-ForStrength*, *Right-ForStrength*, *Left-BackStrength*, *Right-BackStrength*, *UpStrength*, *DownStrength*, *CounterclockStrength*, *ClockStrength*。对于路径中任何两个相邻节点，其连接权值为：

$$T_{ij} = \begin{cases} \text{ForStrength} & A \\ \text{BackStrength} & B \\ \text{LeftStrength} & C \\ \text{RightStrength} & D \\ \text{Left-ForStrength} & E \\ \text{Right-ForStrength} & F \\ \text{Left-BackStrength} & G \\ \text{Right-BackStrength} & H \\ \text{UpStrength} & I \\ \text{DownStrength} & J \\ \text{CounterclockStrength} & K \\ \text{ClockStrength} & L \end{cases} \quad (3-2)$$

式中：*A* 表示机器人向前走一步，而且这一动作是切实可行的，机器人的参考点将从节点 *i* 到达节点 *j*。

B, *C*, ..., *L* 等有与 *A* 类似的定义。

需要注意的是，机器人只能以一种方式从节点 *i* 运动到节点 *j*，因此节点 *i* 和 *j* 的连接权值是唯一的，即 *A*, *B*, ..., *L* 这 12 个条件是互斥的。

(2) 路径长度

对于一个路径规划算法，除了路径的安全性，规划出来的路径的长短也是一个很重要的评价因素。依次将路径中相邻两节点之间的欧氏距离迭加得出整条路径的长度。

综上所述，设计本算法的适应度函数 *f* 为：

$$f = \begin{cases} P_{\max} \times O_{\max} \times T_{\max} - \sum_{i,j} O_{ij} \times T_{ij} & , \text{当个体有效} \\ 0 & , \text{当个体无效} \end{cases} \quad (3-3)$$

式中： P_{\max} 为路径中最大节点数，通常为 8， O_{\max} , T_{\max} 分别是个体中相邻节

点间的欧氏距离和能量消耗的最大权值； i, j 为路径中相邻节点， O_{ij} 为 i, j 两节点间的欧氏距离， T_{ij} 为机器人由节点 i 运动到节点 j 的能量消耗权值。

由此可见，适应度函数的选择考虑了能量的消耗和通路的长度两方面的因素。应用适应度函数评价个体，进行搜索，不仅避免了搜索过程中的“组合爆炸”现象，而且把任务的具体需求信息带进了搜索过程中，使搜索结果令人满意。

3.2.3 遗传参数的设计

遗传算法的参数中交叉概率 P_c 和变异概率 P_m 的选择是影响遗传算法行为和性能的关键所在，直接影响算法的收敛性。 P_c 越大，新个体产生的速度就越快，但如果 P_c 过大时遗传模式被破坏的可能性也越大，使得具有高适应度的个体结构很快就会被破坏；如果 P_c 过小，会使搜索过程缓慢，以至停滞不前。对于变异率 P_m ，如果 P_m 过小，就不易产生新的个体结构；如果 P_m 取值过大，那么遗传算法就变成了纯粹的随机搜索算法。针对不同的优化问题，需要反复实验来确定 P_c 和 P_m ，而且很难找到适应于每个问题的最佳值。

Srinivas 等提出一种自适应遗传算法(Adaptive GA, AGA)， P_c 和 P_m 能够随适应度自动改变。这种调整方法对于群体处于进化后期比较合适，但对于进化初期不利，因为进化初期群体中的较优的个体几乎处于一种不发生变化的状态，而此时的优良个体不一定是优化的全局最优解，这容易使进化走向局部最优解的可能性增加。为此，文献[50]做了进一步的改进，使群体中最大适应度的个体的交叉率和变异率不为零，分别提高到 P_{c2} 和 P_{m2} ，这就相应地提高了群体中表现优良的个体的交叉率和变异率，使得它们不会处于一种近似停滞不前的状态。

计算表达式如下：

$$P_c = \begin{cases} P_{c1} - \frac{(P_{c1} - P_{c2})(f' - f_{avg})}{f_{max} - f_{avg}}, & f' \geq f_{avg} \\ P_{c1}, & f' < f_{avg} \end{cases} \quad (3-4)$$

$$P_m = \begin{cases} P_{m1} - \frac{(P_{m1} - P_{m2})(f - f_{avg})}{f_{max} - f_{avg}}, & f \geq f_{avg} \\ P_{m1}, & f < f_{avg} \end{cases} \quad (3-5)$$

式中: $P_{c1} = 0.8$, $P_{c2} = 0.5$, $P_{m1} = 0.001$, $P_{m2} = 0.0001$

f_{max} —— 群体中最大的适应度

f_{avg} —— 每代群体的平均适应度

f' —— 要交叉的两个个体中较小的适应度

f —— 要变异个体的适应度

3.2.4 模拟退火参数的设计

温度参数是模拟退火算法中的一个关键参数之一。主要包括起始温度的选取、温度的下降方法和停止温度的确定等。

模拟退火算法的中起始温度 T_0 和温度下降方法, 以及停止温度的选择也是影响整体算法行为和性能的关键所在。 T_0 越大或温度下降越慢, 找到全局最优解的概率越高, 但如果 T_0 过大或温度下降太慢, 会使搜索过程缓慢, 算法时空消耗太大, 以至停滞不前, 很难在实际中应用; 如果 T_0 过小, 会使搜索过程过快, 很难找到全局最优解, 失去了算法的优势。对于温度的下降方法, 考虑到本算法的工作环境较大, 对计算机的资源有很高的要求, 所以希望温度的下降方法较为简单, 并易于实现。为此, 本课题用以下的方法设定模拟退火参数。

3.2.4.1 起始温度的选取

起始温度的选区主要思想是为了使平稳分布中每一状态的概率相等, 这样很容易估计一个值为

$$T_0 = k\delta, k \text{ 充分大的数}, \quad (3-6)$$

其中, $\delta = \max\{f_j | j \in D\} - \min\{f_j | j \in D\}$, D 为状态空间。

对一些问题, 有时可以简单地估计 δ , 但有的时候, 会出现 δ 比较难估计或很难精确地知道最大值和最小值而使估计太粗。另外 k 的选取过大会造成计算时间的增加, k 过小则是算法过早陷入局部最优点。

在本课题中, 针对公式 (3-6) 中 $\delta = \max\{f_j | j \in D\} - \min\{f_j | j \in D\}$,

我们可以很容易用程序确定 δ 的值。而通过实验也可以确定 k 的值。在本文中, k 取值为 100。

3.2.4.2 温度下降的方法

温度的下降方法对于非时齐模拟退火算法和时齐模拟算法各有不同。非时齐算法温度下降有固定的形式,但是从实际应用来看,非时齐算法的下降速度太慢,在这里我们只讨论时齐算法的温度下降方法。时齐算法的理论一般要求温度下降到零,整个系统以概率 1 收敛全局最优解。无论直观还是理论要求,温度总是下降的;另外从本课题的实际出发考虑,在大范围的三维工作环境下规划出最有路径需要计算机的大量时空资源,为了节省计算机的时空资源,在这里要用一种比较容易实现的简单方法。因此,两种非常直观的下降方法是:

(1) $T_{k+1} = \alpha T_k$, $k \geq 0$ 且 $0 < \alpha < 1$ 。这种温度下降方法是被应用人员广泛欢迎的方法,它的每一步以相同的比率下降, α 越接近 1 温度下降越快, α 越接近 0 温度下降越慢。

(2) $T_k = \frac{K-k}{K} T_0$, 其中 T_0 为起始温度, K 为算法温度下降的总次数。这

一下降的方法的优点是易于操作,而且可以简单地控制温度下降的总步数。它的每一步下降温度度数相等。

这两种方法都原理简单,易于实现。在本课题中我们选择第一种方法,其中 α 取值为 0.8。

3.2.5 算法的终止原则

遗传模拟退火算法从初始温度开始,通过在每一温度的寻优过程,经过温度的下降,最后达到终止原则而停止。尽管在模拟退火算法中有一定的理论指导,但是终止原则大多是直观的。下面分类讨论。

(1) 零度法

遗传模拟退火的最终温度为零,因而最为简单的原则是:给定一个比较小的正数 ε , 当温度 $T_k \leq \varepsilon$ 时,算法停止。表示已经达到最低温度。

(2) 循环总数控制法

这一简单原则在 3.2.4 节中所讲的温度下降方法(2)中已经提及,即总的温度下降次数为一定值 K ,当温度迭代次数达到 K 时,停止运算。这一原则可以简单有效地控制循环的总次数,容易实现。

(3) 基于不改进规则的控制法

在一定温度及给定的迭代次数内没有改进当前的局部最优解,则停止运算。实际上,模拟退火算法的一个基本思想就是跳出局部最优解,直观的结论就是在一定次数的迭代中没有跳出局部最优解,在下面的迭代中跳出最优解的可能也比较小。由此产生上面的停止原则。

(4) 接受概率控制法

该方法与(3)有相同的思想。给定一个指标 $\chi_f > 0$ 是一个比较小的数,除当前最优解以外,其它状态的接收概率都小于 χ_f 时,停止运算。实现(3)(4)时,记录当前局部最优解,给定一个固定的迭代次数,当在规定的次数里没有离开局部最优解或每一次计算的其它个体接收概率都小于 χ_f ,则在这个时候停止计算。

除此之外,算法的终止原则还有邻域法、Lundy 和 Mees 方法^[61]、Aarts 和 Van Laarhoven 法^[62]等。在本课题中我们采用前三种方法的结合方法:当温度下降到零度的时候,自然要停止循环;同时也使用循环总次数控制法和基于不改进规则的控制法,即达到循环的总次数或当在某个温度及一定的迭代次数内没有改进当前的局部最优解时就认为达到算法终止条件,停止循环。

3.2.6 失败的处理

在算法执行过程中可能会遇到下列三种情况:

- (1) 在一个区域的八个子区域中找不到可以作为子起点的区域的子区域;
- (2) 在一个区域的八个子区域中找不到可以作为子终点的区域的子区域;
- (3) 在一个区域的连通图中找不到连通子起点和子终点的子通路。

当出现上面三种情况之一时,我们说算法执行失败需要进行失败处理。在文献[49]中介绍了一种用失败数组标出造成失败的区域编号,避免下次搜索再选该方法。

这里我们借鉴了失败数组的方法。在对个体进行适应度计算时,当发现

当前个体中出现了失败数组中包括的区域，则将该个体的适应度记为 0，从而该个体在进化中将被淘汰。

3.3 仿真结果

根据本章给出的算法，作者利用 Visual C++ 6.0 对一个三维杆状机器人的全局路径规划进行了仿真实验，选取种群规模为每自由节点 20 个个体，仿真结果如下。

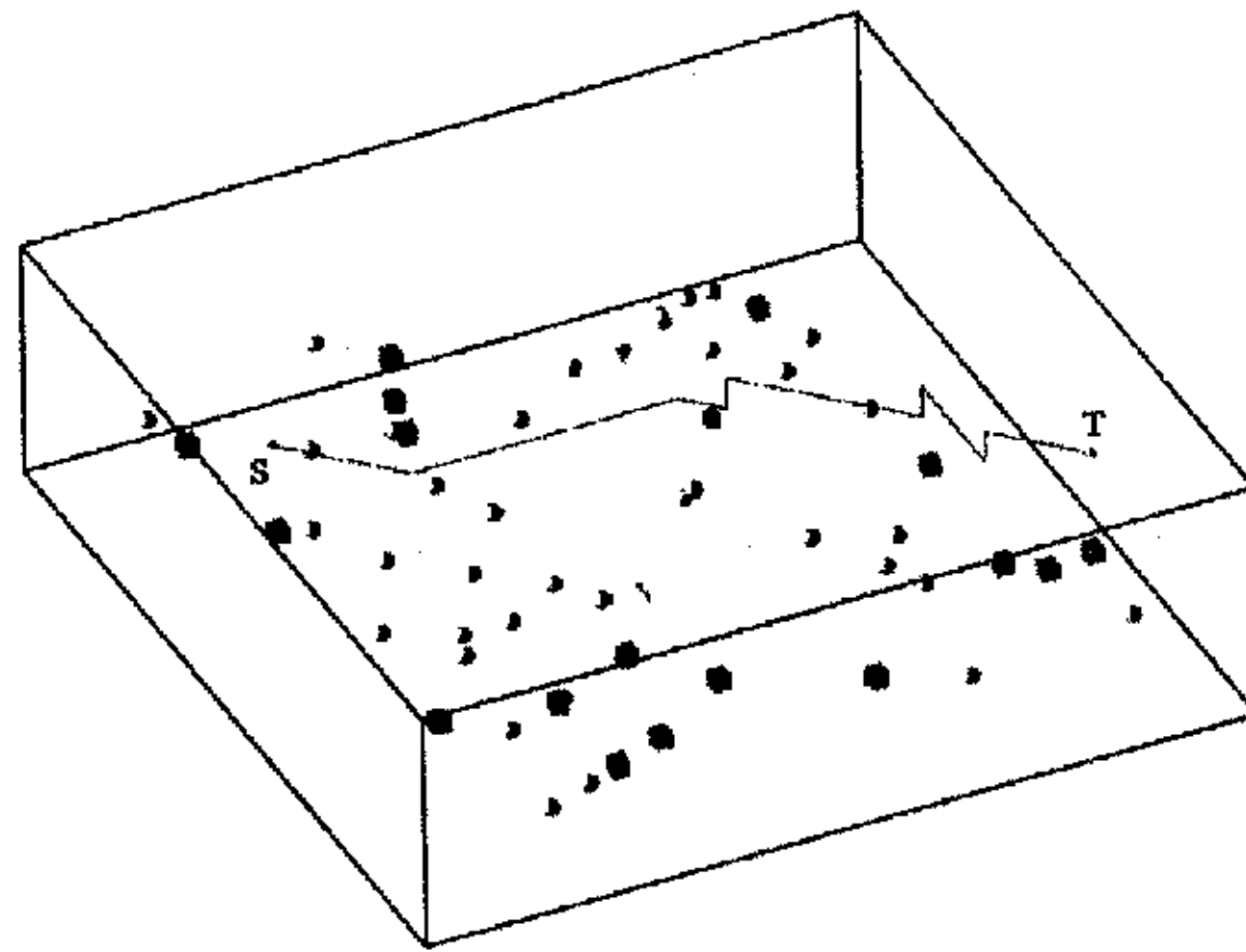


图 3.6 工作环境

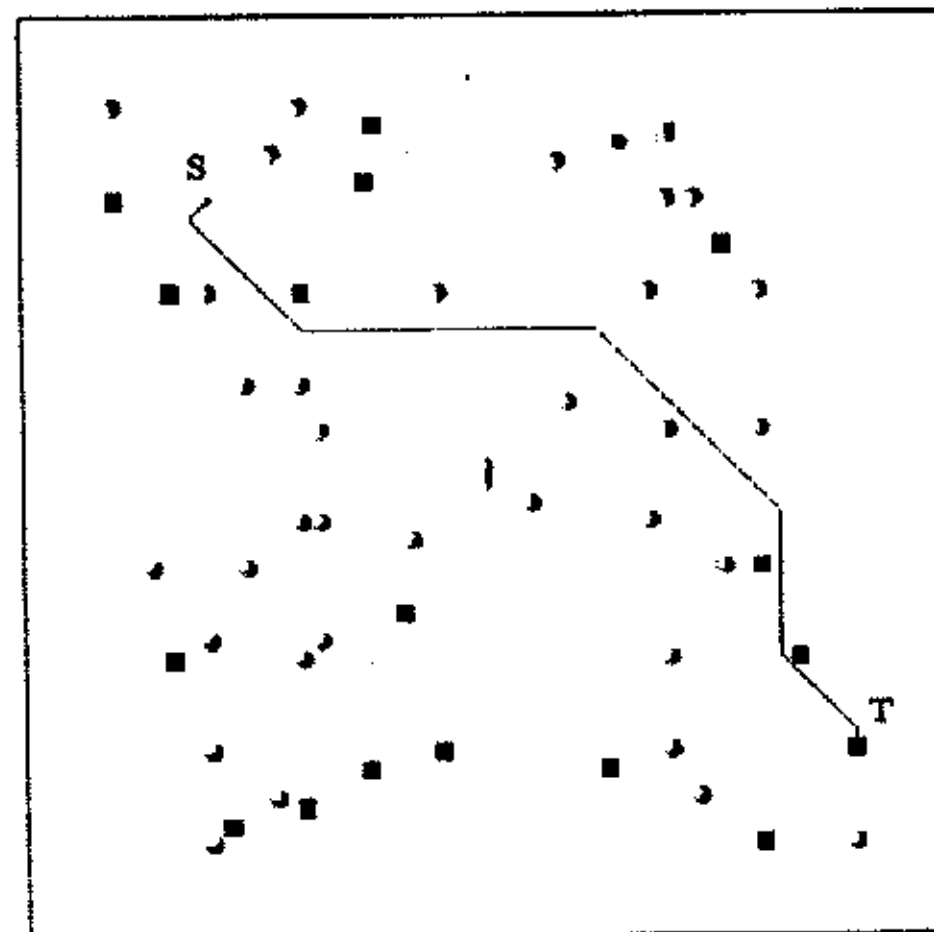


图 3.7 俯视图

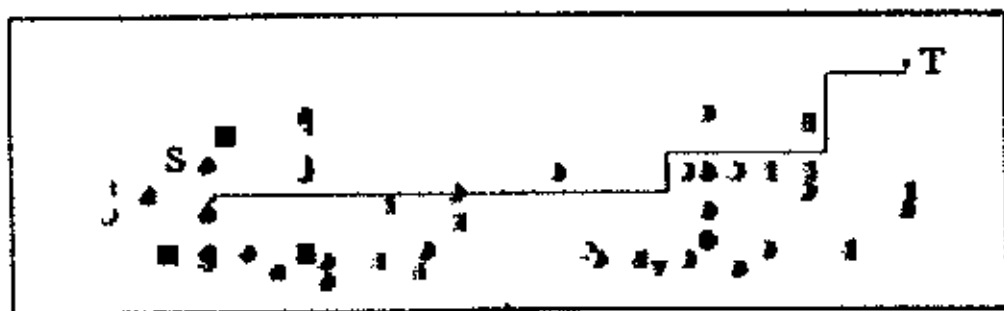


图 3.8 前视图



图 3.9 侧视图

图 3.6 所示的是机器人整个工作环境的概貌,图中的粗线为规划出来的路径。图 3.7 为工作环境的俯视图,图 3.8 为前视图,图 3.9 为侧视图。

3.4 本章小结

本章主要介绍了基于分层模型的遗传模拟退火算法路径规划的基本思想及算法实现中的几个关键技术。

本章首先介绍了区域模型的构造和分层算法的基本思想;然后,详细介绍了搜索算法实现中的几个关键技术:初始种群的设定、适应度函数的设计、遗传模拟退火算法的参数设计、算法的终止原则;最后,给出了该算法的计算机仿真结果。

智能水下机器人所处的环境属于大范围的开放的三维空间,它的复杂性使得规划的难度很大。同时,由于客观环境的限制、实际应用的需要和机器人自身的形态,机器人的位置和运动姿态、求解精度与内存空间之间的矛盾等等也都是路径规划中要考虑的因素。基于分层模型的遗传模拟退火算法路径规划较好的解决了大范围路径规划问题,算法中考虑了机器人行进中的各种能量消耗,经过改进后具有较强的实用性,适合我们智能水下机器人课题的实际应用。该算法把模拟退火的思想与遗传算法相结合,抑制了遗传算法中的早熟现象和陷入局部最优解等缺陷,形成了一种较好的全局路径规划方法。

第 4 章 基于雷达法的局部路径规划

在第 2 章中我们提到机器人路径规划问题可以分为两种，一种是基于环境先验完全信息的全局路径规划，另一种是基于传感器信息的局部路径规划，后者环境是未知或者部分未知的。局部路径规划的工作是在全局路径规划结果的基础上，根据声纳等水底探测设备搜集的实时信息进行判断决策，使得机器人安全的由全局路径的一个关键点走到下一个关键点。

正是因为局部路径规划是根据实时探测到的信息进行规划，所以很适于将避碰、海流等实时问题放到其中来解决。我们这一章所要介绍的基于雷达法的局部路径规划算法就是一种计算简单，符合实时要求，较好的解决了避碰、海流等实时问题的局部路径规划方法。下面就介绍一下算法的基本思想，以及算法实现中的几个关键技术。

4.1 算法概述

基于雷达法的局部路径规划算法是基于栅格模型的一种路径规划方法，它是受启发于雷达的工作原理，具有在未知、部分已知和变化的环境中进行路径规划的能力。由于局部路径规划是在全局路径规划下达的相邻子目标点之间进行的，考虑到如果子目标点之间的距离过大，对大范围的工作区域进行栅格法的表示可能会占用大量的内存空间。因此，本算法针对这一情况，对全局路径规划下达的路径采取按步规划的方法。

在规划之前先设定每次规划的步长 *PlanStep*，即规划起终点之间的距离，然后根据这个值调整全局路径规划下达的路径，使得调整后的路径中相邻两个节点间的距离一定小于或等于规划步长，接下来便在相邻节点之间进行局部路径规划。机器人每次规划的最大步长可以根据机器人的自身特性预先确定，也可以根据规划系统内存大小等实际情况人为进行调整，其值一般可设为几十米到上百米。

在按全局路径导航的过程中，在 AUV 的探测区内有可能探测到新的静

止或运动的障碍物或原地图中现有的障碍物开始运动，称这些障碍物为突发障碍物 SUD ：

$$SUD = O_s \cup O_M \quad (4-1)$$

其中 O_s 为静止的突发障碍物集合， O_M 为运动的突发障碍物集合。其实还有可能探测到目标 T 开始运动，但在本课题中，这方面的问题基本没有涉及，所以我们只作简单讨论。要避开这些突发障碍物和追随运动的目标 T ，就需要进行局部的路径规划。笔者对雷达法进行了合适的应用，用来解决局部动态的规划问题，形成了基于雷达法的局部路径规划算法。

4.1.1 栅格环境模型

对于环境的表示，我们采用了栅格类方法。其基本思想是离散化机器人的工作空间。我们这里规定机器人工作区域为一个以规划起始点 S 为右上角点，目标点 T 为其相对的左下角点的立方体空间，显然该立方体的斜对角线长度等于起始点和目标点之间的距离。由于经过调整后，全局路径中相邻节点之间距离不会超过规划步长，因此确定机器人局部路径规划的步长之后，规划的工作区域大小也就确定下来了。栅格区域的边长可根据下面的公式计算：

$$WorkspaceLength = \sqrt{\frac{1}{3} \times distance_{ST}^2} \quad (4-2)$$

式中： $distance_{ST}$ —— 起始点和目标点之间的距离。

由此，我们可以这样定义路径规划问题，假设 Ψ 为 AUV 的一个 3 维工作空间，将空间 Ψ 划分成 3 维的正方形小格，每一个小格代表一个空间的点 P_i ，那么：

$$\Psi = \bigcup_i P_i \quad (4-3)$$

集合 Θ 包含 Ψ 中所有 AUV 不能到达的立方格，可以通过映射所有的障碍物到 Ψ 求得：在整个栅格空间中，用 0, 1 勾画出障碍物边界，并对障碍物所占的栅格进行填充——无障碍物处的栅格记为 0，有障碍物处的栅格记为 1。所有标记为 1 的栅格点对机器人而言是不可达的，把这些栅格点所构

成的集合就是 Θ ；至于标记为 0 的栅格点是否就是自由的，则与栅格的大小和机器人的规格有关。如果所选取的栅格大小足够的大，以至于对任何标记为 0 的栅格点来说，机器人不论以何种姿态存在，都不会与障碍物相碰，那么所有标记为 0 的栅格点对于机器人来说都是自由的了。但是如果选取的栅格大小小于机器人的最大尺度，则并非所有标记为 0 的栅格点都是自由的。我们把栅格中的自由节点记作 F 。给定一个起始点 S 和一个目标点 T ，路径规划就是要构造一从起点 S 到终点 T 的连接在一起的点序列 Ω ，使得它需要满足条件：

$$\Omega \cap \Theta = \Phi \quad (4-4)$$

显然，如果我们得到了这样的点序列 Ω ，那么它一定满足路径规划中避开障碍物的要求，而我们希望得到的路径不但要满足避碰的要求，还有把距离、位姿、海流等各种能耗因素考虑其中，使得到的路径是一个能耗最小的无碰路径。本文在这里介绍的算法根据机器人配备的前视声纳实时反馈的障碍物信息、全局路径规划下达的机器人路径节点（包括：起始点和目标点）、实时探测的海流方向、速度等条件进行实时搜索，可使机器人由起始节点安全无碰的到达目标节点。

4.1.2 雷达波的传播规则

雷达法来自这样的简单思想：在 Ψ 中给定起终点 S ， T ，终点 T 等时间间隔地发出等速的波沿在 Ψ 中传播，波沿不能穿过障碍物。 S 最先收到波沿的邻点 P 一定是到 T 的理想路径中的一部分，这是因为波沿是朝各个方向等速传播的，所以如果 P 是第一个收到波沿的，它就在一个比其它后收到波沿的点所在路径短的路径上。于是 P 被加入到 Ω 中，并成为新的起点，如果现在 Q 是 P 最先收到波沿的邻点，则 Q 也加入到 Ω 中，并成为新的起点，如此下去，每一个波沿产生一个路径点，直到到达 T ，一条从 S 到 T 的优化路径就形成了。

在这里，我们首先介绍一下雷达波的基本传播规则。图 4.1 显示了波沿在无障碍物的二维工作空间内的传播情况。

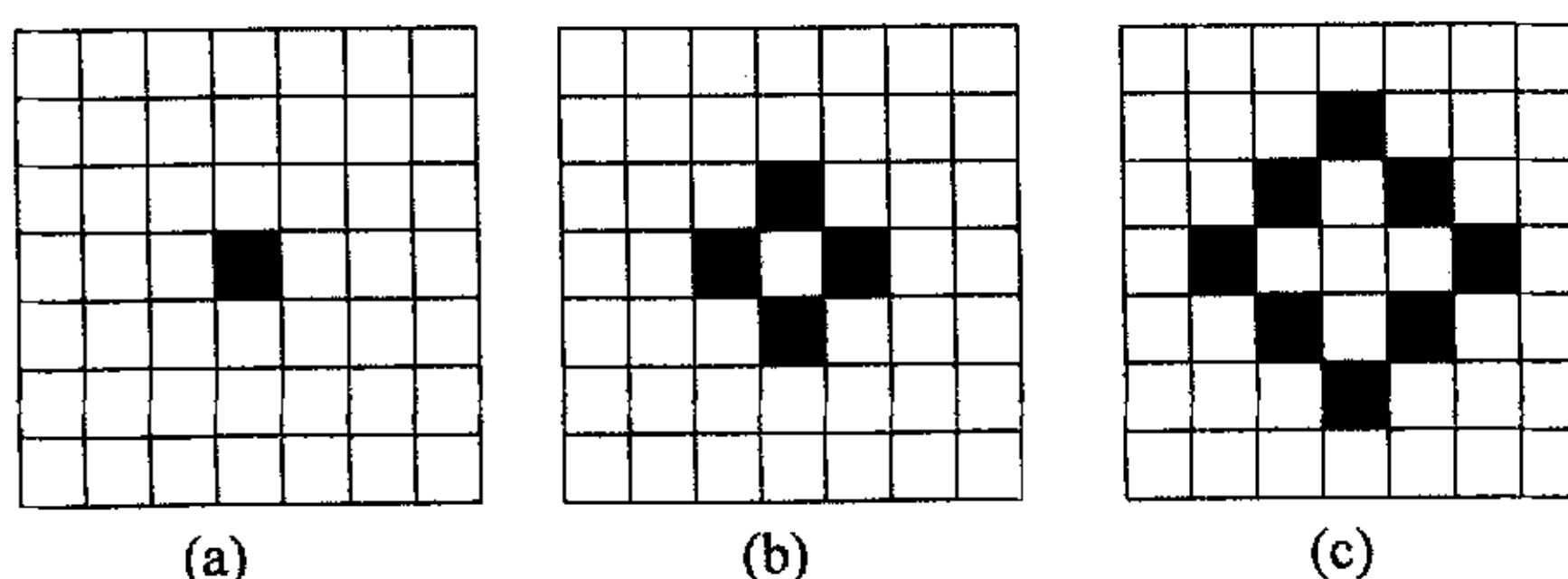


图 4.1 雷达波的传播规则

图 4.1(a)是目标在 t 时激活产生一个波沿的情况, (b)是波沿在 $t+1$ 时的状况, (c)是 $t+3$ 时的情况。一个波沿是 Ψ 中激活点的集合, 在 t 时可通过如下的规则激活每个点 $P \in \Psi$:

$$active(P, t) := \neg obstacle(P) \wedge nhv_active(P, t-1) \wedge \neg active_in(P, t, n) \quad (4-5)$$

其中:

$active(P, t)$: 在 t 时, 当 P 为激活状态返回 true;

$obstacle(P)$: 当 $P \in \Theta$ 时返回 true;

$nhv_active(P, t)$: 在 t 时, 当 P 有一直接邻点为激活状态时返回 true;

$active_in(P, t, n)$: 在 $t-1, \dots, t-n$ 时, 若 P 至少有一次激活过则返回 true。

4.1.3 路径选择规则

一波沿到达起点 S 的邻点时, 某点 P 可以通过如下规则加入到路径 Ω 中

$$path(P) := active(P, t) \wedge cursp_neighbor(P) \wedge (P \neq S) \quad (4-6)$$

其中:

$active(P, t)$: 在 t 时, 当 P 为激活状态返回 true;

$cursp_neighbor(P)$: 当点 P 直接相邻的点被标记为下一次路径选择的起始点时, 返回 true。

如果波最先到达与起始节点 S 邻近的一个节点 P , 那么节点 P 一定处于从起始节点 S 到目标节点 T 的优化路径上。这是因为波在任何方向上的移动速度都是相等的, 所以如果波先到达节点 P , 那么比较那些后来到达的节点, 该节点必定存在一个到达波源的最短路径。将节点 P 加入到路径 Ω 当中并将节点 P 作为新的路径选择起始点, 周而复始。路径就这样一个节点一个节点

地确定直到到达目标点 T 为止。就这样，一条从起始点 S 到目标点 T 的优化路径就这样建立了。

上面只是从原理上介绍路径选择规则，在实际应用中还需要一定的技术，也必须考虑到除距离外其它的一些实际因素如海流、位姿等。

在介绍如何确定位形空间中各节点间的连接权值之前，我们先来说明一下我们的智能水下机器人的运动控制能力，我们在 3.2.2 中讨论过机器人有 12 种运动形式，我们为位形空间中的每个节点定义与其相邻的 12 个节点（环境边缘处的节点除外），并用边把相邻节点连起来。其中在 XOY 平面上有 8 条，分别对应 *Forward*, *Backward*, *Leftward*, *Rightward*, *Left-Forward*, *Right-Forward*, *Left-Backward* 和 *Right-Backward*，同时在 Z 轴的正负方向上各有 1 条边，与 *Upward* 和 *Downward* 对应；从姿态变化的角度来看，从该节点运动到姿态层次比其高（若该节点在第 n 层，则循环连接到第 1 层）而位置相同的节点表示机器人作逆时针方向的旋转（Counterclockwise），从该节点运动到姿态层次比其低（若该节点在第 1 层，则循环连接到第 n 层）而位置相同的节点表示机器人作顺时针方向的旋转（Clockwise）。每条边上都赋予一权值 T_{ij} ，权值的大小表示机器人作相应运动的耗能。对于任何两个都属于自由节点集 F 的相邻节点 $(x_i, y_i, z_i, \theta_i)$ 和 $(x_j, y_j, z_j, \theta_j)$ ，其连接权值 T_{ij} 不为零，表示从一个节点运动到另一个节点的耗能；若两个相邻节点中有一个在障碍物节点集 Θ 中，则它们之间的连接权值为 0。

在本文的算法中路径的选择过程可以分为两个阶段：

第一阶段是雷达波传播的过程，即按照雷达波传播的规律，把局部可探测区域的栅格节点 i 标注出相应的实数值，在这里我们称为节点 i 的激活值，记为 $a(i)$ 。我们把有可能改变其它自由节点激活值的节点称为活动节点。我们在前面提过，因为在规划之前先设定每次规划的步长 *PlanStep*，所以局部目标节点 T 一定在 AUV 的局部可探测区域的栅格中。最初的活动节点是目标节点 T 。从 T 开始，修改与当前活动节点 i 相邻的且满足某些约束条件的自由节点 j 的激活值，并把当前活动节点记为所有这些激活值发生改变的自由节点的父节点，同时将它们作为下一次迭代过程中的活动节点，依次修改与其相邻的且满足约束条件的自由节点的激活值。如此下去，直到所有自由节点的激活值都不再改变为止。对于激活值的初始化可以用(4-7)式来描述：

$$a(i) = \begin{cases} 0 & i \in \Theta \\ \infty & i \in F \text{ 且 } i \neq T \\ 1 & i = T \end{cases} \quad (4-7)$$

上面所说的约束条件可以根据实际情况而定，在我们的应用中所使用的约束条件包括：

(1)与当前活动节点 i 相邻的自由节点 j 的姿态与海流方向的夹角不超过某一允许的角度。这样，若机器人运动到该位姿时，机器人的艏向将满足基本顶流的要求。可以看到，在这里我们很自然地将海流的影响考虑到算法中去。

(2)与活动节点 i 相邻的自由节点 j 当前的激活值 $a(j)$ 大于活动节点 i 的激活值与 i, j 两节点间的连接权值之和，即 $a(j) > a(i) + T_{ij}$ ，则 $a(j) \leftarrow a(i) + T_{ij}$ ，并标记 i 节点为 j 节点的父节点。

(3)当二者相同时，即 $a(j) = a(i) + T_{ij}$ ， $a(j)$ 无需变化，但是可以按照事先约定的条件来决定是否将 i 作为 j 的父节点。这些约定的条件可以和水下机器人的运动控制特性相联系，也可以利用距离启发函数，如选择到目标位姿 *Target* 和起始位姿 *Source* 欧氏距离和最小的邻接节点作为父节点。

第二阶段是按照激活值的梯度方向寻找路径的过程，即形成路径 Ω 的过程。首先把起始节点 S 作为当前搜索节点，从 S 出发，找到当前搜索节点的父节点，并把它作为下一个搜索节点。这个过程不断继续下去，直到目标节点 T 成为当前搜索节点为止。这些搜索节点就构成了从起始位姿 *Source* 到目标位姿 *Target* 的一条耗能最小的无碰路径，同时该路径还具有在小能耗的前提下长度尽量短的特点。最为重要的是该方法保证了当机器人按照该路径行进时，其艏向基本顶流，即艏向与海流方向的夹角在所约定的范围之内。

4.1.4 算法描述

该算法的形式化描述如下所示：

输入：局部栅格区域，起始位姿 S ，目标位姿 T

输出：从起始位姿 S 到目标位姿 T 的路径

部分变量说明：

i, j, k 表示栅格区域中的节点;

$father(i)$ 表示节点 i 的父节点;

QUEUE 表示活动节点队列;

OceanCurrentAngle 表示海流方向;

AnglePermitted 表示所允许的偏移顶流方向的最大角度;

SearchNode 表示当前的搜索节点;

Ω 是用以存放生成的路径的一个队列;

基本步骤:

//步骤 1-4 是对局部栅格区域中的每个节点进行初始化;

步骤 1: 对于栅个区域中的每一个节点 i , 若 $i \neq T$, 做步骤 2 和步骤 3;

步骤 2: $father(i) = \text{Null}$;

步骤 3: 若 $i \in \Theta$, 则 $a(i) \leftarrow 0$, 否则 $a(i) \leftarrow \infty$;

步骤 4: $a(T) \leftarrow 1$, 将 T 放入活动节点队列 QUEUE 中;

//步骤 5-12 是雷达波传播的过程, 即路径选择第一阶段的过程;

步骤 5: 当队列 QUEUE 不空时, 做步骤 6 到步骤 12, 否则转步骤 13;

步骤 6: 取出 QUEUE 中的头元素 i , 作为当前的活动节点;

步骤 7: 对每一个与 i 相邻的节点 j , 若 θ_j 与 OceanCurrentAngle 的夹角不大于 AnglePermitted, 并且 $T_{ij} > 0$ (即 $j \in F$), 则做步骤 8 和步骤 11;

步骤 8: 若 $a(j) > a(i) + T_{ij}$, 则 $a(j) \leftarrow a(i) + T_{ij}$, 转到步骤 10; 否则转到步骤 9;

步骤 9: 若 $a(j) = a(i) + T_{ij}$, 且根据约定条件, 如果 i 要成为 j 的父节点, 则转到步骤 10; 否则转到步骤 7;

步骤 10: $father(j) = i$;

步骤 11: 若节点 j 不在队列 QUEUE 中, 则把节点 j 插入到队列 QUEUE 的尾部;

步骤 12: 转到步骤 5;

//步骤 13-17 是第二阶段, 沿着激活值的梯度方向寻找路径, 即形成

路径 Ω 的过程;

步骤 13: 把节点 S 加入到 Ω 中;

步骤 14: 从 Ω 中取出头元素, 作为当前的搜索节点 $SearchNode$;

步骤 15: 若 $SearchNode = T$, 则结束, 否则转步骤 16;

步骤 16: 把 $father(SearchNode)$ 插入到 Ω 的头部;

步骤 17: 若转到步骤 14;

4.2 算法实现中的关键技术

上一节中介绍了基于雷达法的局部路径规划算法的基本思想及一般过程, 这一节将对算法实现中的几个关键技术进行讨论。

4.2.1 全局路径的调整

前面我们提到了, 在规划之前我们先要设定每次规划的步长 $PlanStep$, 即规划起终点之间的距离, 然后根据这个值调整全局路径规划下达的路径, 使得调整后的路径中相邻两个节点间的距离一定小于或等于规划步长, 接下来便在相邻节点之间进行局部路径规划。

调整全局规划下达的目标点序列的关键在于如何在距离过大的两个路径节点之间插入一个与上一点距离为 $PlanStep$ 的节点。具体寻找分段插入点的方法是: 以点 P 为圆心, $PlanStep$ 为半径作圆, 与线段 PQ 相交于点 R 和 R' 。显然在线段 PQ 延长线上的点 R' 不符合条件, 而该圆与线段 PQ 的交点即为所求分段点 R 。

4.2.2 启发式函数的设定

在路径规划算法中, 适当地使用的搜索策略和启发式函数, 可以使研究的算法得到更好的结果。尽管一般情况下, 在研究算法的时候, 就已经根据环境情况和各种规划要求选择了最好的搜索策略和启发函数, 但是如果使用不同的搜索策略和启发式函数去实现算法, 然后对所得到的结果进行比较, 再从中选择当前最为合理、最优化的路径, 则其效果会更好。

下面我们将主要针对基于雷达法的局部路径规划算法介绍我们在应用中所选择的不同的启发函数。

在 4.1.3 节算法形式化描述中, 步骤 9 的内容是: 若 $a(j) = a(i) + T_{ij}$, 且根据约定条件, 如果 i 将成为 j 的父节点, 则转到步骤 10; 否则转到步骤 7。这一步的意思是: 当一个与活动节点 i 相邻的自由节点 j 当前的激活值 $a(j)$ 等于活动节点 i 的激活值与 i, j 两节点间的连接权值之和, 即 $a(j) = a(i) + T_{ij}$ 时, $a(j)$ 无需变化, 但是要按照一定的条件来决定是否将 i 作为 j 的父节点。这里所说的条件, 其实就是使用启发函数来确定自由节点 j 的父节点是否需要改变。当然, 对于自由节点 j 来说, 如果有多个邻接节点都使得其激活值 $a(j)$ 取得了相同的最小值, 我们可以把它们都作为自由节点 j 的父节点, 然后在网络达到平衡后利用搜索过程最终为 j 确定其中之一为其真正的父节点, 但是这种方法会耗费更多的空间和时间, 从而会影响到规划的速度。另外, 也可以不使用任何启发函数, 而只是从中随机的选择一个。这种方法虽然简单, 但是却有很大的盲目性, 最终可能导致规划出来的路径不能满足能量消耗尽量少、路径尽量短的要求, 甚至在存在路径的情况下得出规划失败的结论。因此这两种方法都不是理想的选择。

如果在算法中使用启发函数则不仅防止了盲目性, 而且会大大减少计算量。我们采用了以下两种启发函数。

(1) 运动形式优先级的启发函数

这种启发函数的选择和水下机器人的运动控制特性有关。对于我们的水下机器人来说, 虽然有 12 种运动方式, 但是各种运动所消耗的能量不同, 因此我们根据所消耗能量的多少, 定义了各种运动形式的优先级, 能量消耗的少, 则相应的运动形式优先级高。下面是各种运动形式按照优先级从高到低的排列顺序: 向前运动, 逆时针旋转和顺时针旋转, 左前运动和右前运动, 向左运动和向右运动, 左后运动和右后运动, 向后运动, 向下运动, 向上运动。由于自由节点间的连接权值反映了从一个节点到另一个节点所消耗的能量, 因此也就反映出各种运动之间的优先关系, 所以通过比较连接权值就可以确定出自由节点 j 的父节点选择哪个。

(2) 距离启发函数

我们所定义的距离启发函数的形式是:

$$f(x, y, z) = distance_0 + distance_1 \quad (4-8)$$

式中: (x, y, z) 是节点 (x, y, z, θ) 的三维空间坐标

$distance_0$ 是 (x, y, z) 到起始节点的欧氏距离

$distance_1$ 是 (x, y, z) 到目标节点的欧氏距离;

如果自由节点 j 有多个邻接节点使其激活值取得最小值, 则利用公式(4-8)计算每个邻接节点到起始节点和目标节点的欧氏距离和, 选择其中欧氏距离和最小的一个作为自由节点 j 的父节点。

从实际应用情况看, 这两种启发函数对路径的规划都有相当大的意义, 有利于选择最优路径。

4.2.3 动态规划策略

我们在本章讨论的是局部路径规划, 是基于传感器信息的路径规划, 机器人的工作环境是未知或者部分未知的。局部的动态路径规划要求避开突发障碍物 SUD , 跟踪目标 T , 还要保持路径尽量最短。在动态的环境中, AUV 按原全局路径航行的过程中必须不断地根据当前探测到的动态环境来实时地进行规划, 任何时刻规划出从当前起点 S 到当前终点 T 的完整路径将是不明智的做法, 因为环境不断地在变化, t 时的完整路径在 $t + \Delta t$ 时很有可能不再满足优化的目标(如障碍物已落在该路径上或已离开, 不再需要绕行)。这个问题可以通过在进行规划时根据实际情况给路径加上一个长度限制的方式来解决, 即每次局部规划只是根据原有的全局路径和当时的环境求出有限长度 len 的路径 Ω_R , 于是选路规则变为:

$$path(P) := active(P, t) \wedge cursp_neighbor(P) \wedge (P \neq S) \wedge (|\Omega_R| < len) \quad (4-9)$$

需要说明的是, 在 4.2.1 节全局路径的调整中提到的, 在局部规划之前把全局规划出的路径分成若干段长度小于或等于步长 $PlanStep$ 的路径, 做这个调整的主要目的是缩小工作空间, 减少内存空间的占用。而这里提到的动态规划时对规划路径加一个长度限制 len 的目的完全不同, 这里的目的是使机器人在动态的环境中, 在避碰的同时尽量找到最优的局部路径。所以 len 的长度选择不能像选择 $PlanStep$ 的长度那么随意, 而是要根据环境变化的速度快慢, 障碍物的多少, 海流的大小等多种因素来确定。

此外，如果在短距离内仍然发生了环境的变化，例如障碍物的移动，并且机器人原有的路径达不到避碰这一基本的要求，即

$$\Omega \cap \Theta \neq \Phi \quad (4-10)$$

在这种情况下，只有马上进行路径的重规划。最直观的方法是重新执行雷达法路径规划，具体做法是：标记 Ω 中 AUV 当前点 P_{cur} 之后所有的路径点为无效，以 P_{cur} 和 T 为边界取一正六面体区域，置 $S_R = P_{cur}$ ， $T_R = T$ ，执行雷达法规划。

除了限定路径的长度 len 外，还有一些动态规划方法也适用在水下机器人的应用环境。例如，一种是当机器人到达关键点时，测当前的海流及其它环境信息；另一种策略是，过一定的时间测一次环境信息，时间间隔可以根据环境变化的速度和特点来确定，该策略的缺点就是机器人除了在关键点需要定位外，还需要根据时间在其它的位置定位，从而减慢了机器人运动的速度。

4.2.4 失败的处理

在算法执行过程中可能会遇到下列两种情况：

- (1) 当前栅格除了刚刚走过的栅格而没有其它栅格可选时；
- (2) 找不到下一个可选栅格时；

当上述两种情况中的任何一种发生时，算法都无法继续执行下去，宣告失败。

当第一种情况出现时，由于当前栅格除了刚刚走过的栅格而没有其它栅格可选，而对于上一个栅格而言当前栅格是最合适的选择，当机器人回到上一栅格后，还会选择刚才的栅格，因此规划执行到这里程序就会陷入死循环。为了避免程序陷入死循环，我们在这里给每个栅格增加一个 *selected* 属性，并规定凡属性 *selected* 被设为 *true* 的栅格不能再参加选择。当一个栅格被发现反复走过 3 次以上，便将它的 *selected* 属性设成 *true*，使其不能再被选中，这样就大大降低了程序在两个栅格间陷入死循环的可能。对于被设为选中的条件，也可以根据算法的需要进行调整，并不一定是 3 次。

当第二种情况出现时，算法不能继续执行下去，这时需要进行必要的回溯处理。造成这种情况的原因主要有两种。一种是由于机器人所处的区域障

碍物十分密集，导致根本找不到安全合适的路径；另一种就是由栅格模型自身的特点造成的，当栅格划分得非常粗，其中每个栅格的面积比较大，则很可能声纳探测时发现周围的栅格都包含障碍物，所以无法选择。这种情况下并不是像第一种情况那样真的找不到路径，如果将栅格划分得更细密一些，其实就能找出不含障碍物的栅格，从而有路可走。图 4.2 给出了二维情况下的示意图。

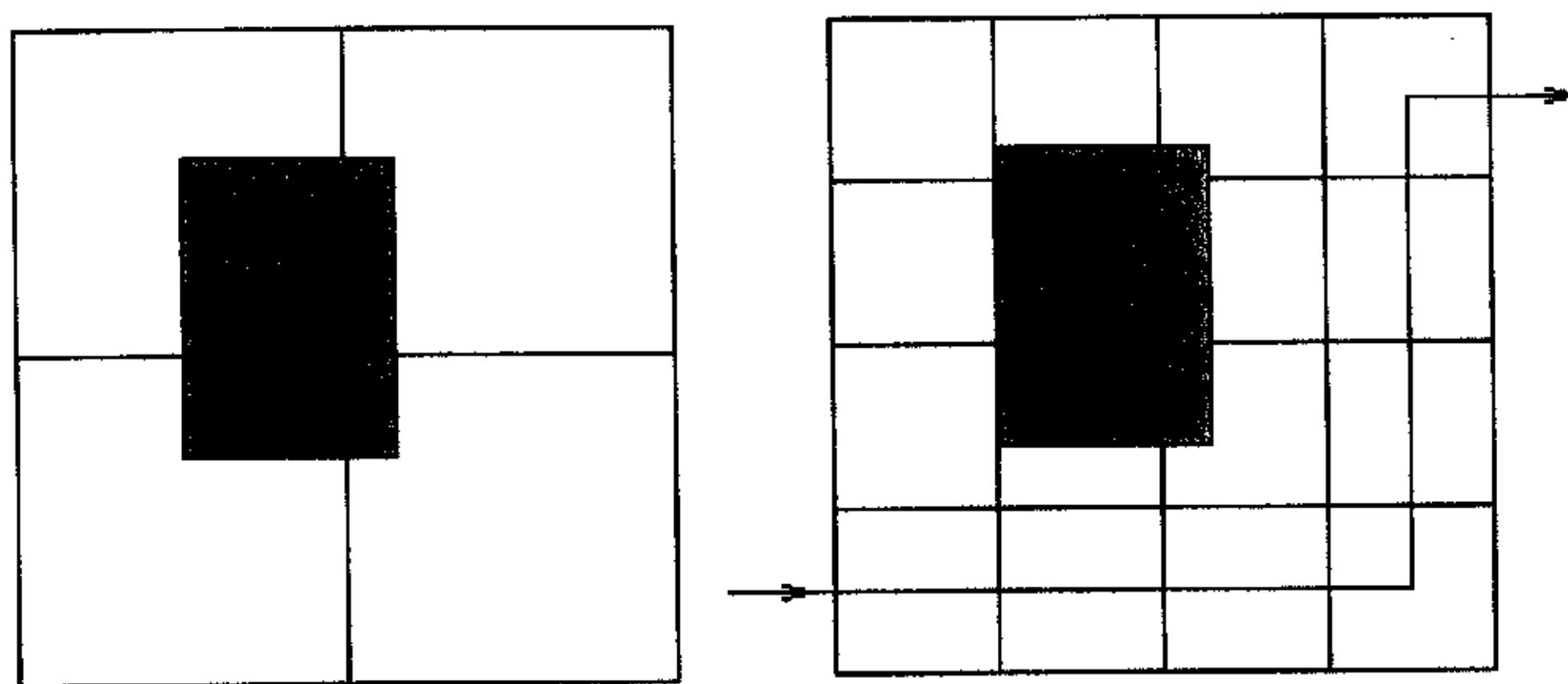


图 4.2 栅格划分对路径规划的影响

理解了栅格划分与路径规划之间的关系，我们可以针对第二种情况作出如下的回溯处理。为栅格区域设置一个回溯路径数组 *back*，在机器人行进过程中，将栅格区域中走过的栅格编号按先后顺序记入 *back* 数组中。当遇到找不到下一个子目标点的情况时，使机器人按走过的路径回退一定数量的栅格。然后适当减小栅格的单位长度，增加区域中栅格的个数，从当前位置重新建立更加细密的栅格模型，再次规划路径。

在规划过程中一旦遇到找不到路径的情况就采取回溯措施，直至找到路径或多次回溯后仍找不到路径，则规划失败。

4.3 仿真结果

根据本章阐述的算法，作者利用 Visual C++ 6.0 进行了仿真实验，仿真结果如下。

图 4.3 所示的是机器人整个工作环境的概貌，图中的粗线为规划出来的路径。图 4.4 为工作环境的俯视图，图 4.5 为前视图，图 4.6 为侧视图。

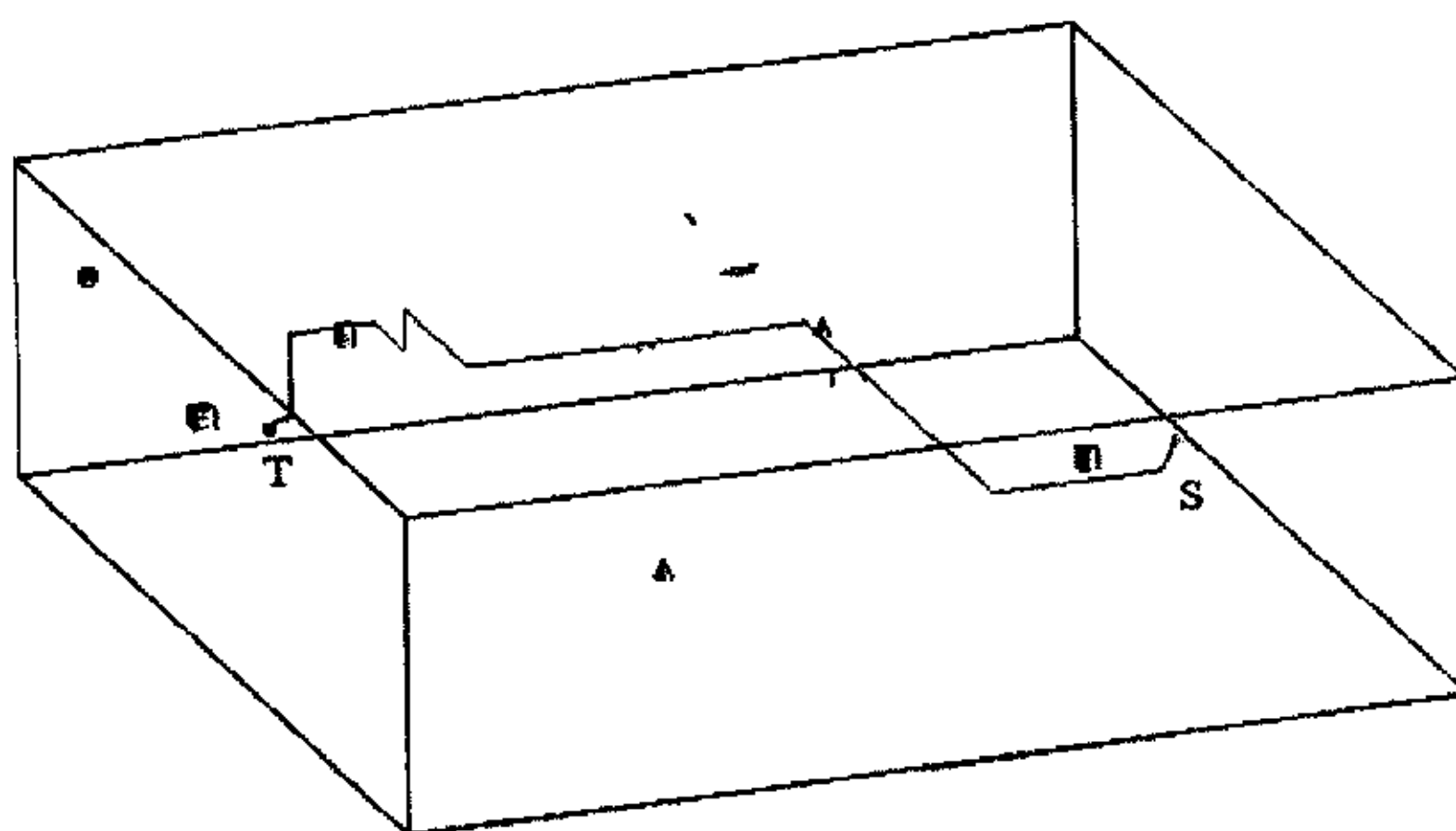


图 4.3 工作环境

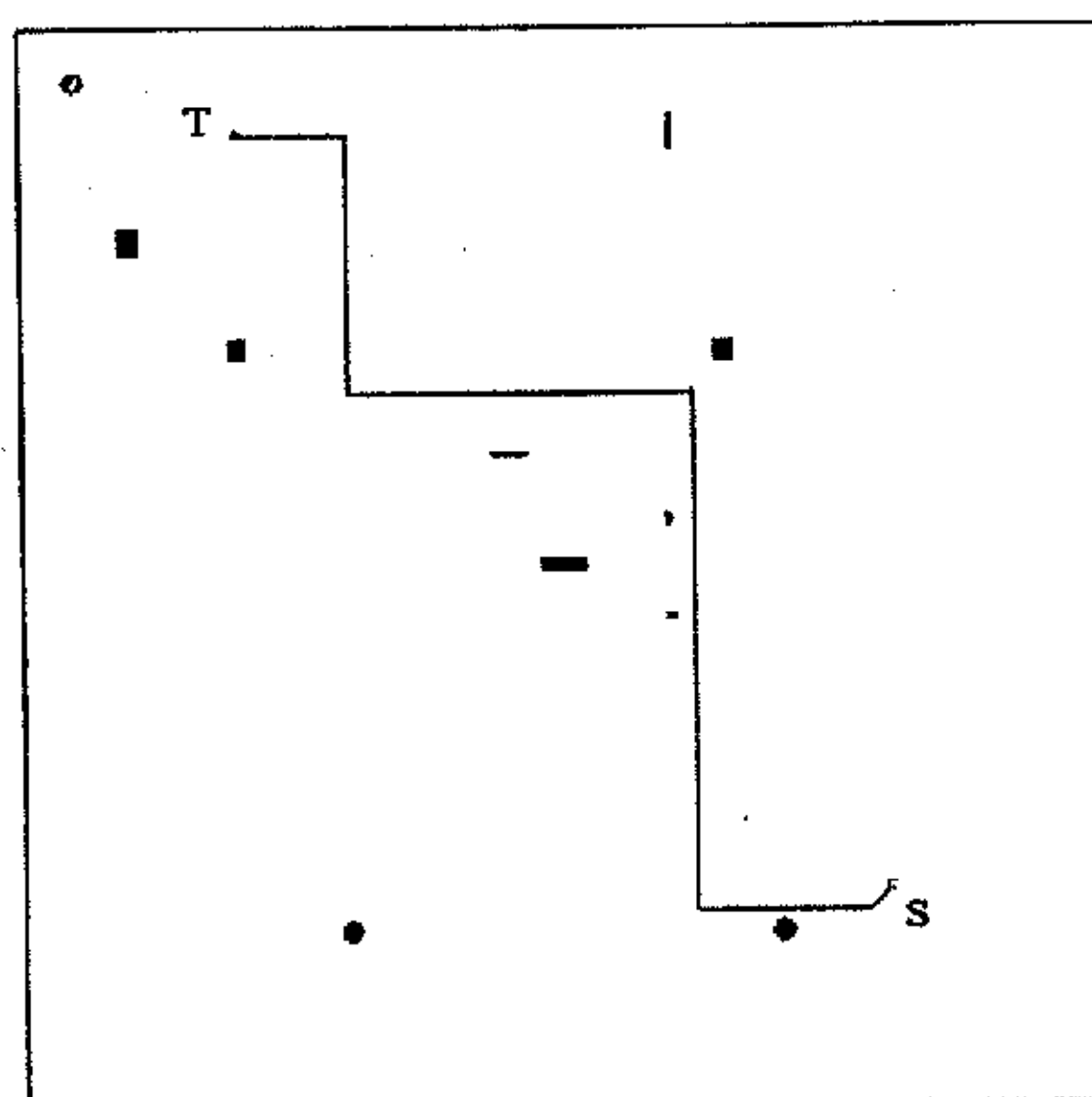


图 4.4 俯视图

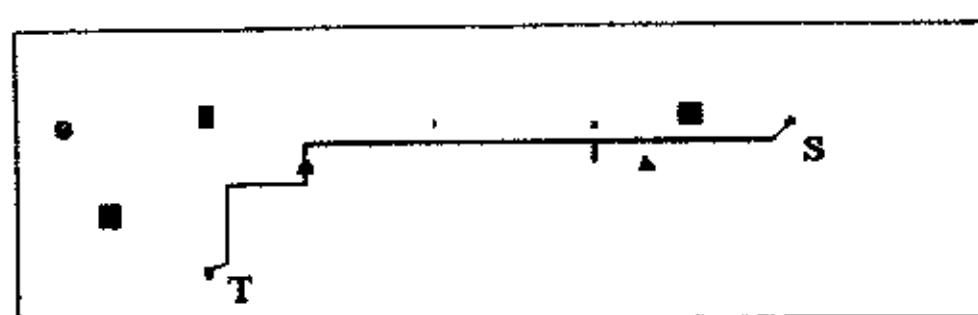


图 4.5 前视图

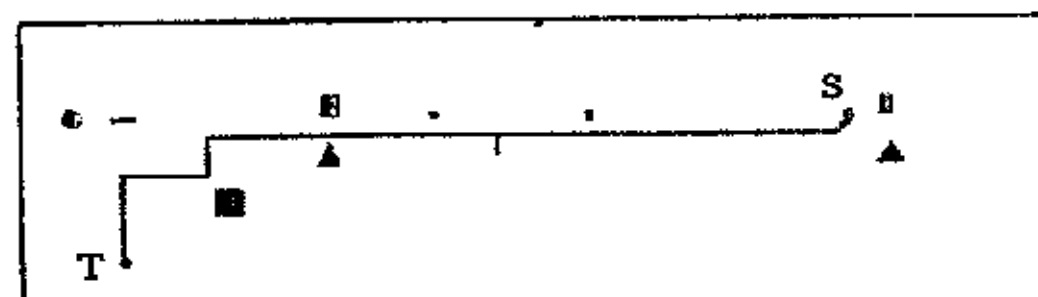


图 4.6 侧视图

4.4 本章小结

本章主要介绍了基于雷达法的局部路径规划算法的基本思想及算法实现中的几个关键技术。

本章首先介绍了算法的基本思想；然后，详细介绍了算法实现中的几个关键技术：全局路径的调整、启发式函数的设定、动态规划策略以及失败的处理；最后，给出了该算法的计算机仿真结果。

基于雷达法的局部路径规划算法利用声纳传感器实时得到的障碍物和海流等环境信息动态调整栅格环境模型，在机器人行进过程中根据利用雷达法对环境模型进行处理，使得机器人能够快速地对环境模型加以判断并决定下一步移动的方案。因此该算法对环境和海流的变化具有快速的反应能力，符合局部路径规划实时处理的要求。仿真结果证明，我们在探讨大范围海洋环境下智能水下机器人的局部路径规划技术方面取得了较好的成果。

结 论

本文对海洋环境下智能水下机器人路径规划技术进行了系统深入的研究,在借鉴了国内外机器人路径规划技术成功经验的基础上,在对水下机器人路径规划方法的理论研究和实际应用过程中,成功的解决了原有算法出现的问题,设计出了一整套适用于海洋环境下机器人的路径规划问题的算法,并给出了全局和局部路径规划的解决方案:基于分层模型的遗传模拟退火全局路径规划方法和基于雷达法的局部路径规划方法。

基于分层模型的遗传模拟退火算法路径规划是一种用于解决大范围全局路径规划的算法,其基本思想是将机器人的工作空间递归地分解为相继层次上的区域,如果该区域完全在障碍区之外,则标记为白区;如果该区域完全在障碍区之内,则标记为黑区;否则,标记为灰区。每层分解后,在白区和灰区中按照遗传模拟退火算法的思想寻找一条连通初始位形和目标位形的最优“通路”,如果得到的通路中含有灰区,则将通路中的区域继续分解为更小的区域,标记它们并重复上面的搜索过程,直到找到一条只含白区的通路,或者确定没有通路存在,或者灰区的尺寸小于给定的精度不能继续分解为止。作者总结遗传算法和模拟退火算法的优缺点,将两种算法结合并应用于机器人路径规划问题,提出了基于分层模型的遗传模拟退火算法路径规划方法。利用模拟退火算法的突跳性,解决了遗传算法局部爬山能力差的问题;另外模拟退火算法的概率接受新的个体的策略保留了个体的多样性,解决了遗传算法交配时丢失最好解的问题和收敛过快的的问题。

基于雷达法的局部路径规划的基本思想受启发于雷达的工作原理。雷达法来自这样的简单思想:在工作环境中给定起点和目标点,目标点等时间间隔地发出等速的波沿在工作环境中传播,波沿不能穿过障碍物。起点最先收到波沿的邻点一定是到目标点的理想路径中的一部分,这是因为波沿是朝各个方向等速传播的,所以如果这个邻点是第一个收到波沿的,它就在一个比其它后收到波沿的点所在路径短的路径上。于是这个邻点被加入到最优路径中,并成为新的起点,如果现在它的某个邻点是最先收到波沿的邻点,

则此邻点也将加入到最优路径中，并成为新的起点，如此下去，每一个波沿产生一个路径点，直到到达目标点，一条从起点到目标点的优化路径就形成了。该方法对环境的表示采用了栅格类方法，因此结合了栅格法和位姿空间的优点。由于将环境信息和机器人运动中能量的消耗编码到节点上和节点间，因此该方法能方便地实现各种优化条件，并适用于各种复杂的环境，具有较好的控制生成路径的能力和可扩展性。

尽管我们在智能水下机器人路径规划方面取得了较好的进展，但是仍有一些方面有待进一步探讨。在基于分层模型的遗传模拟退火算法路径规划中，所使用的遗传模拟退火参数和估价函数对最终生成的路径的影响很大，选择必须慎重；另外，如何改进该算法，提高算法的执行效率、尽可能缩短执行时间也是需要进一步考虑的重要问题。在基于雷达法的局部路径规划算法中，机器人位姿的离散化程度与时空复杂度的矛盾是无法克服的，而且在实际环境中机器人的位置和姿态都分别离散化到什么程度比较合适，也是一个待探讨的问题。

参考文献

- [1] 周远清等. 智能机器人系统. 清华大学出版社. 1989: 1-5 页
- [2] 蒋新松. 机器人学导论. 辽宁科学技术出版社. 1994: 511-516 页, 543-554 页
- [3] N Nilsson. A mobile automation: An application of artificial intelligence techniques. In Proc IJCAI, 1969
- [4] R S Mosher. Test and evaluation of a versatile walking truck. In Proc of the off-road Mobility Res Symp Int Soc Terrain Vehicle Systems, Washington DC. 1968: 359-379P
- [5] 封锡盛, 刘永宽. 自治水下机器人研究开发的现状和趋势. 高技术通讯. 1999, 9(9): 56 页
- [6] 王若兰, 严明. 国外水下机器人发展现状. 机器人情报. 1991, (3): 8-16 页
- [7] 辛建成. 美国海军未来的水下机器人. 机器人技术与应用. 2000, (3): 9-10 页
- [8] 巴士. 自主式水下航行器. 机器人情报. 1992, (4): 20-24 页
- [9] 刘永宽. 未来十年全球无人无缆自主式潜器的发展趋势. 机器人. 1993, 16(3): 185-192 页
- [10] 袁幼零. 水下机器人十年研究与开发: 技术创新项目案例调查报告. 东北科技. 1992, (4): 33-35 页
- [11] 彭慧, 封锡盛. “探索者”号自治式无缆水下机器人控制软件体系结构. 机器人. 1995(3): 177-182 页
- [12] XX.XX.X.X 智能决策和控制技术. 国防科技预研项目开题论证报告. 1997
- [13] XX 智能水下机器人技术“十五”总体方案要点. 2001 13
- [14] 郑南宁. 计算机视觉与模式识别. 国防工业出版社. 1998 年 3 月: 56-57 页
- [15] J. T. Schwartz, M.Shair. A Survey of Motion Planning and Related

- Geometric Algorithms. Artificial Intelligence. 1988, 37(1-3): 157-169P
- [16] 屠大维, 林财兴. 智能机器人视觉体系结构研究. 机器人. 2001, 23(3): 206—210, 233 页
- [17] Howden, W.E. The Sofa Problem. Computer. Vol J.: 299-301P
- [18] Udupa, S. Collision detection and Avoidance in Computer Controlled Manipulators. In proc 5th Int Joint Conf Artificial Intell. Massachusetts Inst. Technol(1987): 737-748P
- [19] J.T.Schwartz, M.Shair. On the “piano Moves” Problem: The Case of a Two Dimensional Rigid Polygonal Body Moving amidst Polygonal Barriers. Pure Appl. Math, 36, 1983
- [20] Loanno-Perez , T. and Wesley , M.A. An algorithm for Planning Collision-Free Paths among Poly-Hedral Obstacle. Common Ass. Computer Math., Vol. ACM-22(1993): 560-570P
- [21] Li Chen Fu, Dong Yueh Liu. An Efficient Algorithm for Finding a Collision-free Path Among Polyhedral Obstacles. Journal of Robotics Systems. 1990, 7(1): 129-137P
- [22] J C Latombe. Robot Motion Planning. Kluwer, Norwell. MA. 1991
- [23] M B Metea. Planning for Intelligence Autonomous Land Vehicles Using Hierarchical Terrain Representatlon. Pro of IEEE Int Conf on Robottes and Automaton. 1987: 1947—1952P
- [24] Ames L.Crowley. Navigation for an Intelligent Mobile Robot. IEEE Journal of Robotics and Automation. 1985, 1(1): 31-40P
- [25] 袁曾任, 姜焕东. 智能移动式机器人的一种全局路径规划方法和基于知识的路径控制器. 机器人. 1992, 14(2): 25-30 页
- [26] R.A.Brooks. Solving the Find-Path Problem by Good Representation of Free Space. IEEE Transaction Systems, Man and Cybernetics. 1983, 13(3)
- [27] A.Elfe. A Senar-based Mapping and Navigation System. Technical Report. The Robotics Institute, Carnegie-Mellon Univ. 1998: 504-512P
- [28] B.Faverjon. Obstacle Acoidance Using an Octree in the Configuration Space of a Manipulator Proc of the IEEE Int. Conf. On Robotics and

- Automation. 1984: 504-512P
- [29] C Alexopoulos, P M Griffin. Path Planning for a Mobile Robot. IEEE Trans on Sys Man and Cybern. 1992, 22(2): 318-322P
- [30] Yahja A, Singh S Stentz. An Efficient On-line Path Planning for Outdoor Mobile Robots. Robotics and Autonomous Systems. 2000, 32(2-3): 129-143P
- [31] Haralick R M. Digital Step Edges from Zero-Crossings of Second Directional Derivatives IEEE Tans on PAMI. 1984, 6(1): 58-68P
- [32] 熊春山, 黄心汉, 王敏. 手眼立体视觉的算法与实现. 机器人. 2001, 23(2): 113-117 页
- [33] 田涛, 潘俊民. 基于散焦图像深度测量的一种新方法. 机器人. 2001, 23(1): 15-19 页
- [34] 崔茂源等. 基于模糊逻辑的自主移动机器人实时滚动路径规划及控制. 吉林工业大学自然科学学报. 1999, 29(1): 58-63 页
- [35] Danny Z. Chen, Robert J. Szczerba, etc. A Framed-Quadtree Approach for Determining Euclidean Shortest Paths in a 2-D Environment. IEEE Transaction on Robotics and Automation. 1997, 13(5): 668-681P
- [36] Yoshifumi Kitamura, etc. 3-D Path Planning in a Dynamic Environment Using an Octree and an Artificial Potential Field. Proceedings of IEEE International Conference on Intelligent Robots and Systems. 1995: 474-481P
- [37] Yunfeng Wang, Gregory S.Chirikjian. A New Potential Field Method for Robot Path Planning. Proceeding of the 2000 IEEE International Conference on Robotics and Automation. 2000: 977-981P
- [38] 孙树栋, 曲严宾. 遗传算法在机器人路径规划中的应用研究. 西北工业大学学报. 1998, 16(1): 79-82 页
- [39] 周明, 孙树栋. 使用遗传算法规划移动机器人路径. 西北工业大学学报. 1998, 16(4): 580-583 页
- [40] 吴晓涛, 孙曾圻. 用遗传算法进行路径规划. 清华大学学报. 1995, 35(5): 14-19 页

- [41] Kimmo Pulakka, Veli Kujanpaa. Rough Level Path Planning Method for a Robot Using SOFM Neural Network. *Robotica*. 1998, 16: 415-423P
- [42] Ulrich Roth, Marc Walker, Arne Hilmann, and Heinrich Klar. TU-Berlin, Institute für Mikroelektronik, Jebensstr. 1, D-10623 Berlin. 1998
- [43] 张汝波, 顾国昌, 张国印. 水下智能机器人模糊局部规划器设计. *机器人*. 1996, 18(3) 158-162 页
- [44] 唐平, 杨宜民. 动态二叉树表示环境的 A*算法及其在足球机器人路径规划中的实现. *中国工程科学*. 2002, 4(9) 50-53 页
- [45] 张汝波, 杨广铭, 顾国昌, 张国印. Q—学习及其在智能机器人局部路径规划中的应用研究. *计算机研究与发展*. 1999, 36(12): 1430-1436 页
- [46] 温文波, 杜维. 蚁群算法概述. *石油化工自动化*. 2002, 1(19) 19-22 页
- [47] 贾巍, 闫丹凤等. 用基于栅格的有限度自由区法解决水下智能机器人路径规划问题. 第二届全球华人智能控制与智能自动化大会论文集. 1997: 456-459 页
- [48] 董笑菊. 海洋环境下智能水下机器人全局路径规划方法研究. 哈尔滨工程大学硕士学位论文. 2000: 40-62 页
- [49] 张巧荣. 智能水下机器人分层全局路径规划方法研究. 哈尔滨工程大学硕士学位论文. 2002: 45-67 页
- [50] 彭艳. 智能水下机器人全局及局部路径规划技术研究. 哈尔滨工程大学硕士学位论文. 2003: 25-49 页
- [51] Kirkpatrick S, Gelatt Jr C D, Vecchi M P. Optimization by simulated annealing. *Science*, 1983, 220: 671-680P
- [52] Liu Yanfei, Qiu Yuhuang. Robot Path Planning Based on Genetic Algorithms with Two-Layer Encoding. *Control Theory and Applications*. 2000, 17(3): 429-432P
- [53] 付岩. 移动机器人路径规划技术的研究. 哈尔滨工程大学硕士学位论文. 1997: 5-7 页
- [54] Brooks R.A, Lazano-Perez. A Subdivision Algorithm in Configuration Space for Finding Path with Rotation. *Proceeding of the 8th International Conference on Artificial Intelligence*. 1983: 799-806P

- [55] David Zhu, Jean-Claude Latombe. New Heuristic Algorithms for Efficient Hierarchical Path Planning. IEEE Transactions on Robotics and Automation.1991, 7(1): 9-18P
- [56] Hou Edwin S.H., Zheng Dan. Mobile Robot Path Planning Based on Hierarchical Hexagonal Decomposition and Artificial Potential Fields. Journal of Robotic Systems.1994, 11(7): 605-614P
- [57] G. Conte , R. Zulli. Hierarchical Path Planning in a Multi-Robot Environment with a Simple Navigation Function. IEEE Transactions on Systems,Man and Cybernetics. 1995, 25(4): 651-654P
- [58] Woong Keun Hyun, Hong Suh. A Hierarchical Collision-Free Path Planning Algorithm for Robotics. IEEE 1995: 488-495P
- [59] Lazano-Perez. Spatial Planning: A Configuration Space Approach. IEEE Transactions on Computers. 1983, 32(2): 108-119P
- [60] 顾国昌, 张巧荣. 移动机器人分层路径规划方法研究. 哈尔滨工程大学学报. 2001, 22(5): 30-34 页
- [61] Lundy M, Mees A. Convergence of an annealing algorithm. Mathematical Programming, 1986, 34: 111-124P
- [62] Aarts E H L, van Laarhoven P J M. Simulated Annealing: Theory and Application. Dordrecht: D Reidel Publishing Company, 1987

攻读硕士学位期间发表的论文和取得的科研成果

- [1] 顾国昌, 付岩. 基于遗传模拟退火算法的水下机器人路径规划. 已投 哈工大学报
- [2] 刘海波, 张国印, 付岩. 面向新闻组的信息搜索方法. 应用科技. 2002, 29(11): 49-51 页

致 谢

本文是在我的导师顾国昌教授的悉心指导下完成的。从大学本科到研究生阶段，顾老师无论是在学习、科研还是在生活方面都给予了我无私的关怀和精心的教诲。他一丝不苟的工作作风，严谨求实、孜孜不倦的治学态度，宽厚仁慈的性格都是非常令人敬佩的，始终是我学习的榜样。正是顾老师的言传身教，使我不仅在学术上，而且在对待生活、对待工作的态度等各方面都受益匪浅。能够在顾老师的指导下完成我的研究生学业，是我一生难得的机遇。在此，我要向顾老师表示我最衷心的感谢！

感谢张国印教授和张汝波教授。在整个论文工作期间，两位老师给予作者诸多的帮助和支持，在此表示深深的感谢。

感谢课题组中曾给予作者协助与支持的刘海波、王醒策、郭轶尊、宋梅萍、陈红洲、王继刚、郑宏琨、杨歌等同学。

在哈尔滨工程大学六年多的学习生活中，作者渡过了最难忘的时光。在此期间，我曾得到了许多老师无私的关怀和热心的帮助与鼓励，使我在各方面得以成长和成熟起来，同时我的朋友武恒也在学习和生活上给予我极大的支持和鼓励。在我的研究生学习生活即将结束之即，一并表示感谢。

最后，我更要深深的感谢我的父母，正是由于他们无微不至的照顾和悉心的培养以及在精神上和经济上给予的强大支撑，才使得我顺利完成学业！