分类号：_____ 密级：_____

ＵＤＣ：_____ 编号：_____

工学硕士学位论文

基于多约束目标的智能水下机器人运动规划方法研究

硕士研究生 ：程俊涵

指 导 教 师 ：孙玉山 教授

学科、专业 ：船舶与海洋结构物设计制造

论文主审人 ： 教授

哈尔滨工程大学

2019 年 3 月

分类号：_____ 密级：_____

ＵＤＣ：_____ 编号：_____

工程硕士学位论文

基于多约束目标的智能水下机器人运动规划方法研究

硕士研究生： 程俊涵

指导教师： 孙玉山　教授

学位级别： 工学硕士

学科、专业： 船舶与海洋结构物设计制造

所在单位： 船舶工程学院

论文提交日期： 2019 年 1 月

论文答辩日期： 2019 年 3 月

学位授予单位： 哈尔滨工程大学

Classified Index:
U.D.C:

A Dissertation for the Degree of M. Eng

Research on Motion Planning of Autonomous Underwater Vehicle Based on Multi-constraint

Candidate: Cheng Junhan

Supervisor: Prof. Sun Yushan

Academic Degree Applied for: Master of Engineering

Specialty: Design and Construction of Naval Architecture and Ocean Structure

Date of Submission: January, 2019

Date of Oral Examination: March, 2019

University: Harbin Engineering University

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）：

日期：　　年　月　日

哈尔滨工程大学

学位论文授权使用声明

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

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

作者（签字）：　　导师（签字）：

日期：　　年　月　日　　年　月　日

摘要

智能水下机器人(Autonomous Underwater Vehicle, AUV)是探索海洋的重要工具，它可以执行地图测绘、环境监测、海底地形测绘、环境评估、管道检查、目标搜索以及水下航行器科学研究等多种任务。对于 AUV 而言，运动规划能力是其智能的重要体现，它贯穿 AUV 工作的始终，是 AUV 的重要组成部分，因此对运动规划技术的研究具有重要而深远的意义。

本文研究并设计了一种基于多约束目标的运动规划系统，使用深度强化学习方法，综合考虑 AUV 的传感器限制以及其执行机构的约束，实现了 AUV 在无地图环境下，躲避障碍物同时抵达目标点的运动规划任务。论文的主要工作如下：

（1）本研究以哈尔滨工程大学开发的某小型水下机器人为基础，参考其传感器配置与执行机构能力，同时考虑并使用仿真测试了 AUV 的操纵性能，对运动规划系统进行分析及建模。为完善 AUV 运动规划系统，构建了基于 S 面控制方法的控制器，并对控制器进行了仿真试验，验证了其可行性。

（2）本文针对传统强化学习模型难以处理连续动作空间的问题，设计并实现了一种基于策略的 AUV 运动规划系统，使用深度强化学习方法直接逼近策略，优化策略，实现了 AUV 的连续动作空间的规划，可以达到更为精细的规划效果。除此之外，针对 AUV 的运动规划任务需求，参考课程学习的思想，设计了适用于 AUV 运动规划训练的课程。让规划系统在完全的未知环境中进行了仿真测试，验证了系统的可行性。

（3）本文针对深度强化学习奖励函数难以设计，容易出现意外解以及连续状态动作空间下奖励稀疏等问题，设计并实现了一种基于好奇心奖励的奖励函数设计方法。改方法模拟了人类的好奇心，鼓励机器人更多的探索位置的环境状态，训练过程说明了其在更大的状态空间下，好奇心奖励的优势。与此同时，让规划系统在完全的未知环境中进行了仿真测试，验证了系统的可行性。

关键词：智能水下机器人；运动规划；深度强化学习；好奇心奖励

ABSTRACT

Autonomous Underwater Vehicle (AUV) is an important tool for exploring the ocean. It can perform various tasks such as map mapping, environmental monitoring, seabed top mapping, environmental assessment, pipeline inspection, target search, and underwater vehicle scientific research. For AUV, the ability of motion planning is an important embodiment of its intelligence. It runs through the AUV and is an important part of AUV. Therefore, the research on motion planning technology has important and far-reaching significance.

This paper studies and designs a motion planning system based on multi-constraint. Using the deep reinforcement learning method, considering the sensor limits of AUV and the constraints of its actuators. In a mapless environment, the AUV can avoid obstacles and simultaneously reach the target's motion planning. The main work of this paper is as follows:

(1) Based on a small underwater robot developed by Harbin Engineering University, this study refers to its sensor configuration and actuator capabilities, and uses simulation to test the operational performance of AUV, and analyzes and models the motion planning system. In order to improve the AUV motion planning system, a controller based on the S-plane control method was constructed, and the controller was simulated to verify its feasibility.

(2) The traditional reinforcement learning model is difficult to deal with the continuous action space. In this paper, a policy-based AUV motion planning system is designed and implemented. The deep reinforcement learning method is used to directly approximate the strategy and optimize the strategy to realize the continuous motion space planning of AUV. A more detailed planning effect can be achieved. In addition, for the AUV motion planning task requirements, with reference to the idea of curriculum learning, designed a curriculum for AUV motion planning training. The planning system was tested in a completely unknown environment to verify the feasibility.

(3) It is difficult to design the deep reinforcement learning reward function, because of easy to have unexpected solutions and reward sparseness in continuous state action space. This paper design and implement a reward function based on curiosity reward. The improvement method simulates human curiosity and encourages robots to explore the unknown environmental states. The training process illustrates the advantages of curiosity

reward in a larger state space. At the same time, the planning system was tested in a completely unknown environment to verify the feasibility of the system.

Keyword: Autonomous Underwater Vehicle, Motion planning, Deep reinforcement learning, Curiosity reward

目录

第1章 绪论

1.1 研究背景与意义

海洋拥有丰富的海洋生物资源，矿产资源和能源。它是人类可持续发展的重要资产。研究和合理利用海洋对人类的经济和社会发展具有重要意义。如今，海洋的开发和勘探对所有国家都极具吸引力和挑战性。海洋的恶劣环境使人类难以征服海洋，而水下机器人是人类了解海洋，开发和利用海洋的主要载体之一。

水下无人机器人中，智能水下机器人（Autonomous Underwater Vehicle，简称 AUV）将多个学科（如：人工智能、自动控制、模式识别、信息融合与理解、系统集成等技术）应用于水下平台上，能够自主的执行工作任务，是水下机器人技术目前发展的趋势[1]。AUV 具有很大的应用潜力，而且在国际领域的应用正在逐步增加。在海洋研究中，AUV 可用于海底测绘和长期环境监测（如温度，盐分，深度测量，海洋污染测量等）、水下结构测量（如海底输油管道检测，海上平台水下结构监测，海底电缆检查和铺设等）。世界上所有主要军事强国都非常重视它们在未来战争中的应用。作为一个水下无人平台，它们可以应用于情报收集，目标识别，扫雷和反潜工作等。AUV 将成为未来水下战争的竞争优势，在战场上实施精确打击和智能攻击以及完成特种作战任务的重要手段之一。

智能水下机器人在未来主要向大型化和微小型化两个方向发展[2]。大型智能水下机器人具有高负载能力和持久耐用的特点，并配备有各种传感器，可以进行多种水下操作。例如用于铺设海底光缆的 Theseus 型智能水下机器人、4500m 级海下作业平台海马号 ROV、由沈阳自动化研究所开发的基于探索者型号的新型机器人 CR-02 型以及由哈尔滨工程大学开发的智水 IV 等。而微小型机器人则具有体积小，灵活，成本低，易于部署和回收等特点。例如由美国 Oceanographic Systems Lab (OSL)设计，Hydroid 公司开发和生产的 REMUS 100，由韩国 Korea Ocean Research and Development Institute (KORDI)开发的 ISIMI，以及由英国 Heriot-Watt University Ocean Systems Laboratory 开发的 Nessie VT。

AUV 需要自主的完成任务，因此其运动规划技术的相关研究具有重要的实际意义。运动规划贯穿了 AUV 水下航行的始终，是其完成水下作业任务的基础。由于海洋环境复杂多变，已知的环境信息不一定十分准确，所以需要 AUV 可以在未知环境中进行运动规划以保证 AUV 水下航行的安全。本课题基于某小型水下机器人进行了研究，实现了其在未知环境下的运动规划，这是水下机器人顺利完成其他任务的基础，研究依托于

装备预先研究：动态环境下的行为规划技术，具有重要的意义。

1.2 小型水下机器人研究现状

智能水下机器人（AUV）具有广阔的应用前景，其中小型机器人由于其体积小、阻力小、操作灵活等优势，在其技术的研究和开发中给予了极大的关注和投入。迄今为止，美国在小型水下机器人领域的研究进展较为先进，但是其他国家对于小型水下机器人的研究同样给予了极大的关注和投入。

（1）REMUS 100-S

REMUS 是由美国 Oceanographic Systems Lab (OSL)设计，Hydroid 公司开发和生产的一系列水下机器人。该系列是世界上最著名的水下机器人系列之一[3][4]。

图 1.1 REMUS 100-S

REMUS 100-S 长 1.84m，直径 0.19m，空气中质量 45kg，最大工作深度 100m，其最大前进速度 2.6m/s，额定速度 1.54m/s，续航时间约为 10 小时，外壳使用铝合金制造。REMUS 100-S 配置有 Kongsberg 惯性导航系统、精密 GPS、声通讯等设备，可以额外安装测深系统、多波束、侧扫声纳、摄像机等设备。可以用于执行水文调查、前后疏浚调查、渠道监控、管道检查、紧急响应调查、搜索和恢复操作、快速环境评估以及水下机器人科学研究等任务。

（2）Iver2

Iver2 是由 OceanServer Technology 开发的小型水下机器人[5]。Iver2 长 1.26m，直径 0.15m，空气中质量 19kg，最大工作深度 100m，最大前进速度 2.06m/s，额定速度 1.29m/s，续航时间约为 12 小时，外壳使用铝合金制造。Iver2 配置有 3DOFs 罗盘、深度传感器、GPS 以及高度计，可以额外安装侧扫声纳以及多波束声呐等设备。可以用于海滩调查、沿海地图绘制、环境监测、淡水地图测绘、海洋科学调查、海洋调查、快速环境评估、科学研究、海底测绘以及水下机器人研究等任务。

图 1.2 Iver2

（3）Tethys

Tethys 是由 Monterey Bay Aquarium Research Institute 开发生产的一款远程自主水下机器人(long-range AUV, LRAUV)。Tethys 长 2.3m，直径 0.31m，空气中质量 110kg，最大前进速度 1m/s，额定速度 0.5m/s，额定速度下续航时间约为 740 小时，外壳由铝合金制造。

这种新型远程 AUV 的可移动范围和耐久性极大地扩展了自主平台可以执行的观测和实验类型。Tethys 扩展了研究者的无船舶观测范围，同时也提供了持续两周至一个月的持续观测能力，是进一步海洋研究的基础。

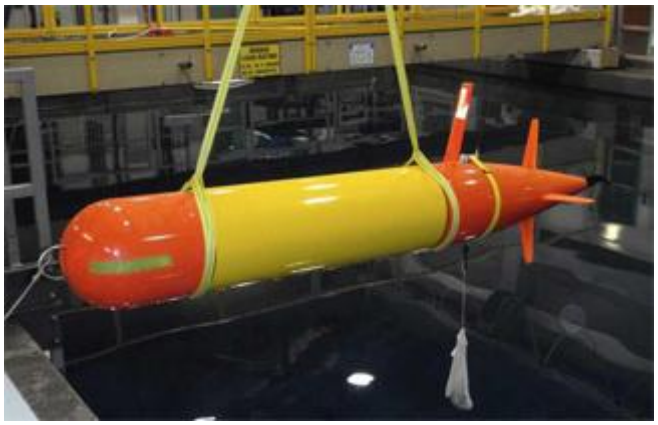


图 1.3 Tethys

（4）Nessie VT

Nessie VT 是由 Heriot-Watt University Ocean Systems Laboratory 开发的小型水下机器人。Nessie VT 长 1.6m，直径 0.28m，空气中质量 40kg，最大工作深度 100m，最大前进速度 2.6m/s，额定速度 1.5m/s，额定速度下续航时间约为 22 小时，外壳由铝合金制造。Nessie VT 配置有 6 个高功率推进器可以控制 5 个自由度的运动。Nessie VT 配备了一系列最先进的传感器，这些传感器包括多普勒测速仪（DVL），两个前视声纳，四

个摄像机，温度传感器，压力传感器，GPS 和光纤陀螺仪。Nessie VT 具有悬停功能，主要用于竞赛、科学实验以、一般海底研究以及标准调查任务。

图 1.4 Nessie VT

（5）Tri-Dog 1

Tri-Dog 1 是由 University of Tokyo Institute of Industrial Science 开发的小型自主水下机器人[6]。Tri-Dog 1 长 1.85 米，直径 0.58m，空气中质量 170kg，最大工作深度 100m，额定速度 0.72m/s，额定速度下续航时间约为 3 小时，外壳由铝合金制造。Tri-Dog 1 配备六个推进器，可独立控制 4 个自由度的运动，光纤陀螺、多普勒速度记录、姿态航向参考系统以及深度计。Tri-Dog 1 配置有四个避障声纳以及三个摄像机。Tri-Dog 1 是作为浅水试验台开发的，主要用于靠近建筑物时的操作任务。

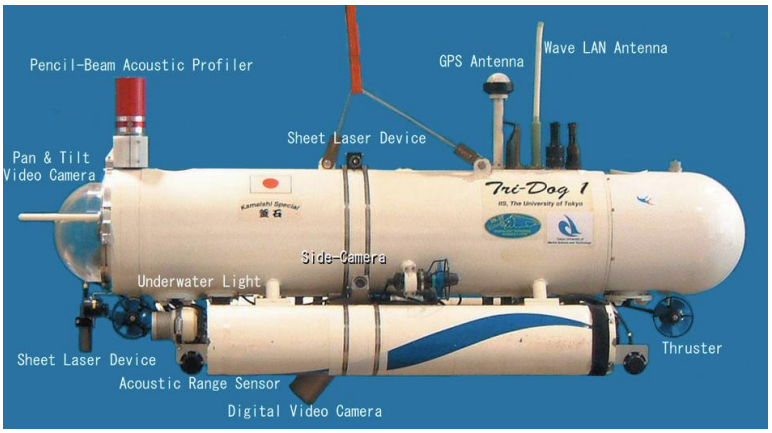


图 1.5 Tri-Dog 1

（6）Gavia

Gavia 是由 Teledyne Gavia 设计生产的一款模块化小型智能水下机器人。Gavia 长 1.7m，直径 0.2m，空气中质量 49kg，最大工作深度 1000m，最大前进速度 3m/s，额定速度 1m/s，额定速度下的续航时间根据配置不同在 5-7 小时之间，外壳由铝合金制造。

模块化的设计可以让用户根据需求选择 Gavia 需要配置的传感器。海滩调查、沿海地图测绘、环境监测、淡水地图测绘、地球物理调查、海洋科学调查、矿产调查、海洋调查、快速环境评估、科学研究、海底地形测绘、水下机器人科学研究等多种任务。

图 1.6 Gavia

（7）微龙-I 型水下机器人

微龙-I 是哈尔滨工程大学开发的一种小型水下机器人[7]。它长 0.95m，空气中质量为 76kg，排水量 80kg，最大工作深度 50m，额定航速 1m/s，额定速度下续航大约 4-5 小时。微龙-I 是一款扁圆形的智能水下机器人，外壳由 FRP 制造，耐压舱使用铝合金制造。微龙-I 配置有摄像机、声纳以及深度计等多种设备，具有自主航行以及自主探测的能力。

图 1.7 微龙-I

1.3 水下机器人运动规划研究现状

运动规划智能水下机器人的重要组成部分。能水下机器人的运动规划是根据某些评估标准在已知或未知环境中找到从初始状态到目标状态的运动轨迹。在未知环境中，运动规划是一个复杂的多约束问题。海洋地理环境中的障碍以及水下机器人自身执行器的局限性对运动规划的安全性有很大影响。因此，忽略这些约束并简单地强调到达目标点的运动规划方法在实际应用中是不可行的。

综合考虑水下机器人自身执行机构的约束以及海洋障碍物的约束等多约束目标，构建一个基于多约束目标的水下机器人未知环境运动规划方法是很有必要的。这可以通过传感器获取环境、目标点以及机器人机身的信息，实时生成及修正轨迹来实现。未知环境的运动规划系统结构如图 1.8 所示。

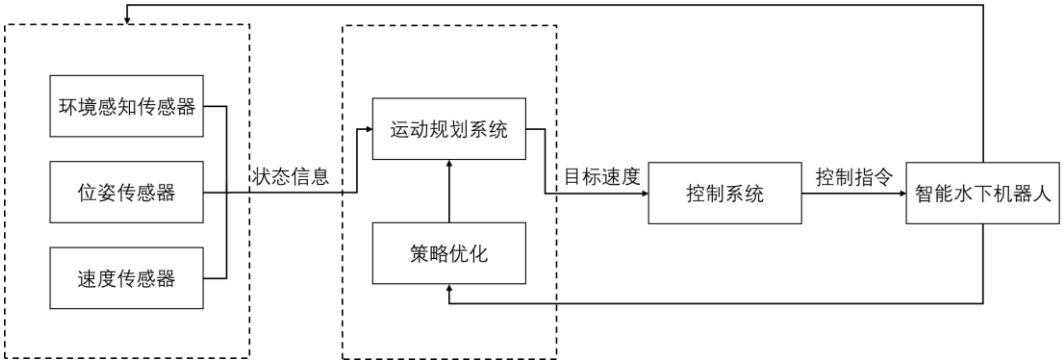


图 1.8 智能水下机器人运动规划系统结构

AUV 的运动规划算法主要可以归纳为 4 大类：基于图的搜索方法、基于采样的方法、基于人工势场的方法、基于人工智能的方法，除此之外还包括各种方法的改进与融合，如图 1.9 所示。

（1）基于图的搜索方法。

基于图的搜索方法根据已知的障碍物地图按照一定的最优策略进行搜索，最终获得起点至终点可行的轨迹解。但是这种方法规划出的轨迹解通常并不平滑，无法满足水下机器人的操纵性。常用的基于图的搜索方法包括可视图法、Dijkstra 算法以及 A*算法。

可视图法首先根据已知的障碍物地图构建可视图，然后按照一定的最优策略进行搜索路径，这种方法可以求得最优路径，但是搜索时间过长，而且只可以应用于多边形障碍物，在存在圆形障碍物时算法失效，对于可视图法，有切线图法以及 Voronoi 图法等改进算法。

Dijkstra 算法是一种广度优先的搜索方法，这种方法可以确保第一个轨迹解即最优路径，但是效率低下。A*算法是对 Dijkstra 算法的一种改进算法，它在 Dijkstra 算法的基础上加入了启发函数，改进了 Dijkstra 算法效率低下的问题，是最为常用的基于图的路径搜索算法。

陈超等人提出了一种基于可视图方法的水面无人艇运动规划方法，首先使用可视图构建海洋环境模型，结合 A*算法进行路径搜索，实现 USV 的运动规划[8]。Garau 等人提出了一种基于 A*算法的水下环境运动规划方法，在环境的构建过程中考虑了流场的影响，实现了水下的节能规划[16]。Stentz 等人提出了一种基于 D*算法的 AUV 运动规划

方法，通过优化 D*算法的关键参数，实现 AUV 在动态环境中运动规划[20]。

（2）基于采样的方法

基于采样的路径规划方法主要指快速搜索随机树（Rapidly-exploring Random Tree, RRT）算法。RRT 算法快速搜索环境，并且在规划的过程中可以考虑水下机器人的操纵性约束。与此同时，作为一种随机搜索算法，RRT 在高维空间中也有着很好的适应性。但是，RRT 算法由于是随机搜索，因此算法效率很低，而且算法只具备概率完备性。对于 RRT 算法，具有很多种改进算法，其中最为重要的是 RRT*算法以及滚动规划，RRT*算法是渐进最优的，滚动规划是一种将 RRT 用于局部路径规划的改进方法。

Hu 等提出了一种基于采样的移动机器人运动规划方法，可以应用于多智能体协同规划[9]。Yang 等提出了一种基于增量采样的移动机器人运动规划方法，可以解决具有非线性动力学约束的运动规划问题[10]。庄佳园等提出了一种基于改进随机树算法的水面无人艇运动规划方法，在 RRT 算法中加入抑制因子，可以兼顾水面无人艇的操纵性能以及规划路径的最优性[11]。

（3）基于人工势场的方法。

人工势场是一种虚拟力法，其基本思想是将障碍物与目标点编辑为势场，障碍物势能高，目标点势能低，势能差产生了虚拟力，障碍物对机器人产生排斥力，目标点与机器人产生吸引力，其合力控制机器人抵达目标点，同时躲避障碍物。人工势场计算简单，执行效率高。但是，人工势场存在着局部最优的问题。对于人工势场方法，主要有两种改进方向，一是改进势函数以避免出现局部最优解；二是改进搜索策略在机器人陷入局部最优解后离开局部最优解，最为常用的策略有模拟退火、随机搜索等。

在水下机器人的运动规划领域，王芳等人提出了一种基于改进人工势场的智能水下机器人运动规划方法，对人工势场进行的构建方法进行了改进，以解决局部最优解问题[12]。李欣等人提出了一种基于改进人工势场的智能水下机器人运动规划方法，对势函数进行了改进，以解决局部最优解问题[13]。Warren 等人提出了一种基于人工势场的智能水下机器人运动规划方法，在算法的基础上增加了启发性的数据，以解决局部最优解问题[18]。

（4）基于人工智能的方法。

基于人工智能的路径规划方法是指将现代人工智能技术应用于路径规划中，不特指具体的某一种方法。人工智能方法包括进化算法以及人工神经网络方法等，这些方法都可以应用于水下机器人的路径规划中，它们各自具有不同的优缺点。

进化算法包括遗传算法(Genetic Algorithm, GA)、蚁群算法(Ant Colony Optimization, ACO)、粒子群算法(Particle Swarm Optimization, PSO)等，这些算法都模拟了生物的行为，通过不断的进化迭代寻找最优路径，这些方法都设计了可以跳出局部最优解的结构，但是需要大量的搜索帮助算法收敛。在智能水下机器人的运动规划领域，进化算法有着很多的应用，刘利强等实现了一种基于蚁群算法的，应用于水下机器人三维环境运动规划方法，但是并没有改进蚁群算法本身存在的收敛速率过慢以及陷入局部最优解的问题[23][24]。徐玉如等提出了一种基于粒子群优化与遗传算法的混合算法的，考虑海洋海流环境影响的水下机器人二维运动规划方法[21]。Alvarez 等人提出了一种基于进化算法的 AUV 运动规划方法，考虑了三维环境以及复杂还留的影响[17]。Yuh 等人提出了一种基于遗传算法的 AUV 运动规划方法，并进行了实验，试验结果表明无论在二维还是三维的水下环境中，遗传算法都可以帮助机器人规划出有效路径[19]。毛玉峰等人提出了一种基于粒子群算法的 AUV 运动规划方法，在算法中加入了适应度函数，降低了 AUV 的能耗，同时考虑了 AUV 的操纵性能对算法的影响[22]。

人工神经网络方法是一种模拟人类大脑神经元的方法，人工神经网络可以轻松地处理复杂的问题模型以及约束条件，但是它需要依赖大量的数据进行神经网络参数的训练。俞建成等提出一种基于神经网络的水下机器人自适应运动控制方法[14]，彭良等提出了一种基于人工神经网络的运动控制方法，它将滤波与控制进行了良好的结合，仿真试验验证了其方法的可行性[15]。

近年来，随着计算机技术的发展，智能机器人的自主能力已经成为研究热门。在当前的各种运动规划方法中，将智能机器人的感知与决策相结合，现在被认为是提升智能机器人的自主能力最有效的办法。人工神经网络作为一种人工智能方法，它具有很强大的感知能力，但是在决策领域，其并不具备优势。强化学习是一种通过不断的试错，积累经验进行决策的方法。强化学习在决策方面具有一定的优势，张汝波[25]、杨广铭[26]、李江浩[27]、Moore[28]，Vamvoudakis[29]等人已成功将强化学习应用于完成移动机器人的运动规划任务。

将强化学习与神经网络相结合，是目前人工智能的研究热点，也被认为是未来的发展方向。深度强化学习方法作为将强化学习与神经网络相结合的一种方法，由 Silver 于 2013 年被提出[30][31]。这种方法已经在水下机器人的控制领域进行了应用，Cui R 使用深度 Q 学习实现了水下机器人的自适应控制[32]。但是在运动规划领域的应用还很少见，作为强化学习的改进方法，它可以改进之前基于强化学习的水下机器人运动规划方法所

面临的维度灾难问题，有很大的研究价值。

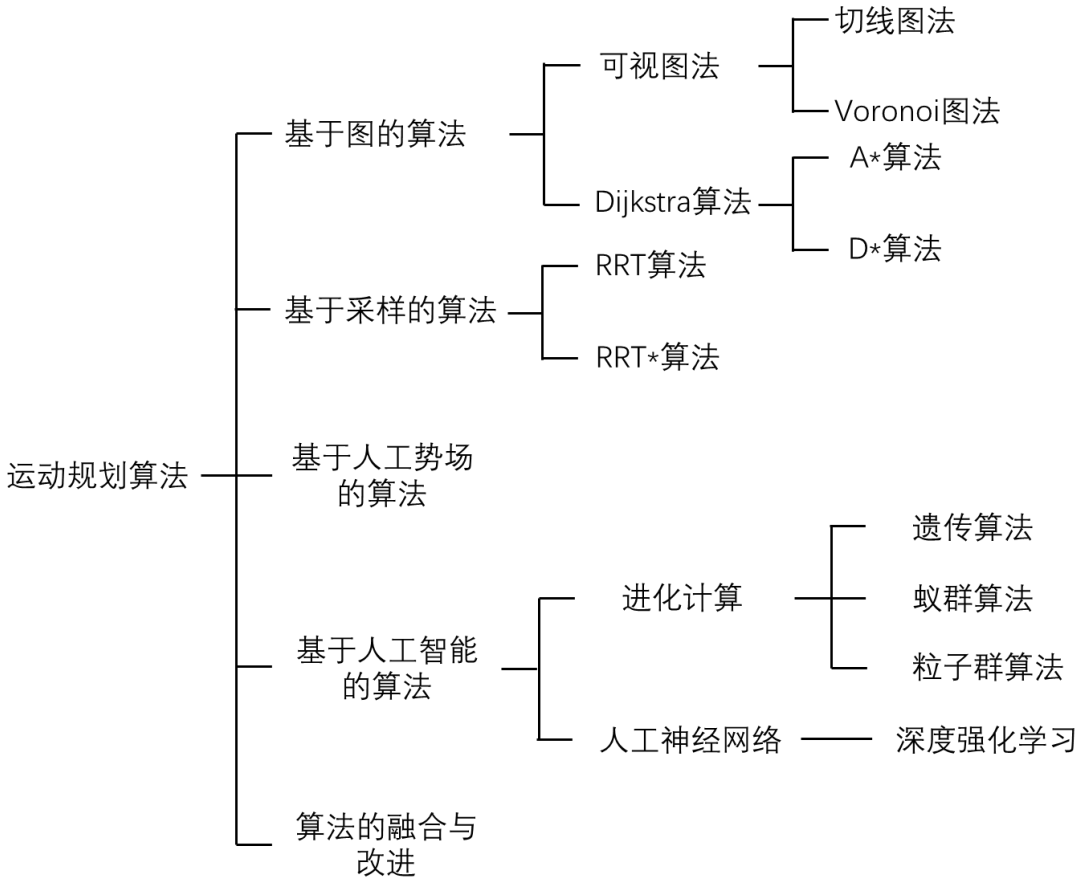


图 1.9 常见运动规划算法

1.4 本文研究内容

本文主要研究小型水下机器人在未知环境下的运动规划问题。它存在以下问题：1）小型水下机器人体积小，可以安装的传感器设备受到限制，可以获得的信息存在局限性，这对运动规划模块的鲁棒性有着很高的要求；2）本课题基于的某小型水下机器人在水平面上只具有两个向前的推进器，没有侧向推进器，是一个欠驱动的 AUV，执行能力有限，对运动规划模块的灵活性有着很高的要求；3）本课题研究的是未知环境下的运动规划问题，因此需要假设 AUV 对环境没有任何了解，完全依赖避障声呐获取环境障碍信息，这对运动规划模块的实时性有着很高的要求。

针对以上问题，本文展开了深入研究，主要研究内容如下：

（1）介绍了水下机器人动力学模型，以及水下机器人的系统组成。分析了未知环境下水下机器人运动规划的多约束目标，建立了基于多约束目标的水下机器人运动规划模型。介绍了深度学习与强化学习的理论背景以及研究现状。

（2）基于深度强化学习的理论基础，针对水下未知环境的特点，提出了一种水下机器人的运动规划方法。方法使用传感器信息作为规划器的输入，目标速度作为规划器的输出，结合 S 面控制方法，实现了水下机器人的运动规划系统，并搭建了仿真试验平台。

（3）为了解决基于连续动作空间的强化学习模型奖励稀疏的问题，提出了一种课程训练的强化学习训练方法。基于课程学习的思想，将训练分步进行，对水下机器人的抵达目标点与躲避障碍物的能力分开训练，以加快算法的训练速度。于仿真试验平台中进行了仿真试验。试验证明在未知环境中算法也可以完成运动规划任务。除此之外，与基于地图规划的 A*算法进行了对比。

（4）基于课程学习的训练方法，本文对深度强化学习模型的训练方法进行了进一步的改进，提出了一种基于好奇心奖励的奖励塑造方法，以改善深度强化学习会获得局部最优解的问题。与无好奇心奖励的模型轨迹进行了对比。试验证明了方法可以避开局部最优解。

第2章 深度强化学习方法

2.1 强化学习算法理论

2.1.1 强化学习发展现状

强化学习的研究有着悠久的历史。目前常用的强化学习算法包括蒙特卡罗、Q 学习、SARSA 学习、TD 学习、策略梯度和自适应动态规划等。表 2.1 中对强化学习的方法发展现状进行了总结。

表 2.1 强化学习发展现状

1956	Bellman 提出了动态规划方法[33]
1977	Werbos 提出自适应动态规划方法[34]
1988	Sutton 提出了 TD 算法[35]
1992	Watkins 提出了 Q 学习算法[36]
1994	Rummery 等提出了 SARSA 学习算法[37]
1996	Bertsekas 等提出了解决随机过程优化控制的神经动态规划方法[38]
1999	Thrun 提出了部分可观测马尔科夫决策过程中的蒙特卡罗方法[39]
2006	Kocsis 等提出了置信上限树算法[40]
2009	Lewis 等提出了反馈控制自适应动态规划算法[41]
2014	Silver 等提出确定性策略梯度算法[42]

在运动规划的领域，强化学习也有着非常多的应用。Kawano 等提出了一种基于强化学习的运动规划方法，实现了欠驱动 AUV 在强水流环境下躲避障碍物抵达目标点[43]。Carreras 等提出了一种基于强化学习的行为控制体系结构，实现了水下机器人目标跟踪[44][45]。Andres 等提出了一种基于策略梯度强化学习的控制方法，实现了 AUV 的轨迹跟踪[46][47]。强化学习在其他领域的运动规划问题上也有很多研究，Lei 等提出一种基于强化学习的运动规划方法，实现了移动机器人避障[48]。Chris 等提出了一种基于强化学习的运动规划方法，实现了机械手 7 自由度运动[49]。Andrew 等提出了一种基于强化学习的运动规划方法，实现直升机原地盘旋[50]。

2.1.2 马尔可夫决策过程

在强化学习中，马尔可夫决策过程（Markov decision process, MDP）描述了一个完全可观察的环境，即观察状态的完整性决定了决策所需的特征。几乎所有强化学习问题都可以转化为 MDP，这可以说是强化学习问题的理论基础。

马尔可夫决策过程是一个满足马尔可夫属性的无记忆随机过程。马尔可夫属性意味

着在环境中，只要知道当前状态，就可以在不依赖历史信息的情况下确定下一个状态，这可以描述为

$$p_{ss'}^{a} = \mathbb{P}[S_{t+1} = s' | S_t = s, A_t = a] \tag{2-1}$$

马尔可夫决策过程可以由集合$< S, A, P, R, \gamma >$表示，其中S表示有限数量的状态集，A表示行为集合，P代表示状态转移概率矩阵

$$P = \begin{bmatrix} p_{11} & \cdots & p_{1n} \\ \vdots & \ddots & \vdots \\ p_{n1} & \cdots & p_{nn} \end{bmatrix} \tag{2-2}$$

其中n是状态数，矩阵中每行元素的总和为 1; R表示概率函数，令R_s表示S状态的奖励，意味着在时刻t，代理在s状态执行动作a，转换为$t+1$时刻的状态s'可以获得的奖励的期望，$R_s^a = E[R_{t+1} | S_t = s, A_t = a]$；$\gamma \in [0,1]$表示衰减系数(Discount Factor)。

定义G_t是马尔可夫奖励过程中从时间t开始的所有奖励的衰减之和，表示为：

$$G_t = R_{t+1} + \gamma R_{t+2} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \tag{2-3}$$

其中衰减系数反映了在当前时刻未来奖励的价值比率。在时间$k+1$获得的奖励R在时刻t由$\gamma^k R$表示，并且γ接近0，表示它倾向于“近视”性评价；γ接近1表示偏好长期利益。

定义策略π，它是概率的集合，其元素$\pi(a|s)$是在过程中在某一状态s采取可能的行为a的概率。表示为：

$$\pi(a|s) = \mathbb{P}[A_t = a | S_t = s] \tag{2-4}$$

定义基于策略的行为价值函数$v_\pi(s)$，表示按照当前策略从状态s开始所获得的收获的期望；或者说在执行当前策略π时，衡量个体处在状态s时的价值大小。表示为：

$$v_\pi(s) = \mathbb{E}_\pi[G_t | S_t = s] \tag{2-5}$$

定义基于策略的行为值函数$q_\pi(s,a)$，该函数表示在执行策略π时，对当前状态s执行特定行为a能得到的收获的期望。或者，当遵循当前策略π时，测量对当前状态执行行为a的收获。表示为：

$$q_\pi(s,a) = \mathbb{E}_\pi[G_t | S_t = s, A_t = a] \tag{2-6}$$

马尔可夫决策过程的状态值函数和行为值函数可以通过下一时刻的状态值函数和下一时刻的行为值函数来表示，即 Bellman 期望方程（Bellman Expectation Equation）：

$$v_\pi(s) = \mathbb{E}_\pi[R_{t+1} + \gamma v_\pi(S_{t+1}) | S_t = s] \tag{2-7}$$

$$q_\pi(s,a) = \mathbb{E}_\pi[R_{t+1} + \gamma q_\pi(S_{t+1}, A_{t+1}) | S_t = s, A_t = a] \tag{2-8}$$

定义最优状态值函数$v_*(s)$，这意味着在所有策略生成的状态值函数中选择最大化状态s价值的函数；类似地定义最优行为值函数$q_*(s,a)$，这意味着在由所有策略生成的行为值函数中，选择状态行为对(s,a)价值最大的函数。

对于任何 MDP 问题，始终存在一个确定性最优策略，可通过最大化最优行为值函数找到：

$$\pi_*(a|s) = \begin{cases} 1 & if\ a = \text{argmax}_{a\in A}\, q_*(s,a) \\ 0 & otherwise \end{cases} \tag{2-9}$$

可以看出，如果知道最优行为价值函数，就找到了最优策略。对于v_*，状态的最优价值等于从状态获取的所有动作生成的行为值中的最大行为值；对于q_*，在某个状态s下，采取行为a的最优值由两部分决定，一部分是离开状态s的直接奖励，另一部分是所有可达状态s'的最优状态值期望，即 Bellman 最优方程（Bellman Optimality Equation）：

$$v_*(s) = \max_a q_*(s,a) \tag{2-10}$$

$$q_*(s,a) = R_s^a + \gamma \sum_{s'\in S} P_{ss'}^a v_*(s') \tag{2-11}$$

通过求解 Bellman 最优方程可以得到最优策略，但 Bellman 最优方程是非线性的，没有固定解。本文通过一种基于策略迭代的方法来解决。

2.1.3 基于策略迭代的强化学习算法

强化学习中最重要的方法之一是 Q 学习（Q learning）方法，Q 学习是一种基于价值的强化学习方法。它需要每一个状态行为对的行为价值，然后根据这些价值选择价值最大的动作。当状态名称无法区分或用于描述状态的特征限制了状态的完美描述时，个体获得的状态信息等同于部分观察的环境信息，则问题不会具有马尔可夫属性。此时，最优策略将不再是确定性的。为了解决这个问题，本文采用策略学习的方法来解决这个问题。基于策略梯度的强化学习方法是通过不断优化策略直接求解策略的一种方法，与基于价值函数的强化学习方法相比，优点如下：

（1）基于策略的学习可能有更好的收敛性。基于值函数的学习方法总是在后期围绕最优价值函数震荡而不收敛；

（2）基于策略的学习在解决连续空间问题时效率更高，如果行为空间维度较高或者是连续的，选取价值函数最高的行为这个过程十分困难；

（3）基于策略的学习可以学习一些随机策略，但基于价值函数的学习通常不会学习随机策略；

（4）基于策略的学习计算更为简单。某些情况下计算价值函数非常复杂。

策略迭代学习直接参数化策略本身。参数化策略不再是概率集，而是一个函数：

$$\pi_\theta = \mathbb{P}[a|s,\theta] \tag{2-12}$$

策略函数π_θ确定在给定状态和特定参数设置的情况下采取任何可能行为的概率，因此实际上它是概率密度函数。当实际应用策略生成行为时，根据该概率分布执行行为采

样。

设$J(\theta)$代表任何类型策略的目标函数，可以是依照策略π_θ可以获得的平均奖励。优化策略的目的即尽可能获得更多的奖励，梯度上升算法通常来讲是最为优秀的优化算法。策略梯度算法即帮助$J(\theta)$沿着其梯度上升至局部最大值。同时可以确定$J(\theta)$最大值时的参数：

$$\theta_{new} = \theta_{old} + \sigma\nabla_\theta J(\theta) \tag{2-13}$$

其中σ是步长参数，又称学习率；$\nabla_\theta J(\theta)$是策略梯度，表示为：

$$\nabla_\theta J(\theta) = \begin{pmatrix} \frac{\partial J(\theta)}{\partial \theta_1} \\ \vdots \\ \frac{\partial J(\theta)}{\partial \theta_n} \end{pmatrix} \tag{2-14}$$

基于策略迭代的强化学习算法的最终由公式（2-13）$\theta_{new} = \theta_{old} + \sigma\nabla_\theta J(\theta)$ (2-13 寻找最优策略。

2.2 深度学习算法理论

2.2.1 深度学习发展现状

深度学习起源于人工神经网络。首先提出的是受到大脑皮层工作方式启发的多层感知机，并且针对多层感知机实现了一种 BP 算法进行优化。但是由于受到当时计算机硬件的条件限制以及算法自身的缺点，深度学习的发展陷入了瓶颈。

随着计算机硬件的飞速发展，计算资源的飞速提升，深度学习取得了重大进展。与此同时，预训练[51]方法的提出，大幅提高了深度学习算法的训练速度以及稳定性。深度学习在语音识别[52]、机器视觉[53]以及目标检测[54]等领域的应用取得了巨大突破。

Krizhevsky 等提出了一种深度卷积神经网络(convolutional neural network, CNN) AlexNet，在数据集 ImageNet 上的图像识别错误率降低至 37.5%，远远低于之前的方法[55]。Graves 等提出一种基于长短时记忆(long short terms memory, LSTM)的语音处理方法，相比传统的递归神经网络有着更好的效果[56]。近五年来，深度学习在很多领域都取得了突破性进展，Xu 等人提出了一种注意力(attention)[57]深度学习模型。Pinheiro 等人提出了一种循环神经网络与卷积神经网络相结合的 RNN-CNN[58]深度学习模型。He 等人提出了一种深度残差[59]深度学习模型。

在运动规划领域，深度学习也有着很多应用。Lecun 等提出了基于深度学习的运动规划方法，将单目视觉图像直接映射到角度偏差，实现越野机器人避障[60]。Chen 等提

出了一种基于深度学习的运动规划方法，对图像信息进行特征提取，根据特征进行自动驾驶[61]。Pfeiffer 等提出了一种基于卷积神经网络（CNN）的运动规划方法，将图像信息直接映射为的动作，实现陆地机器人避障[62]。在水下的运动控制领域深度学习也证明了其有效性[63][64]，深度神经网络多被应用于补偿水动力系数，完善水下机器人运动控制系统。

2.2.2 神经元与感知机

神经网络是由自适应简单单元组成的广泛并行互连的网络，其组织可以模拟生物神经系统与现实世界对象的相互作用。1943 年，McCulloch 和 Pitts 将生物大脑皮层中的神经工作模型抽象为一个简单的模型，如图所示，即“M-P 神经元模型”[65]。在该模型中，每一个神经元接收其他神经元输出的信号，这些信号经过加权处理。然后通过“激活函数”（activation function）处理比较以产生神经元的输出。

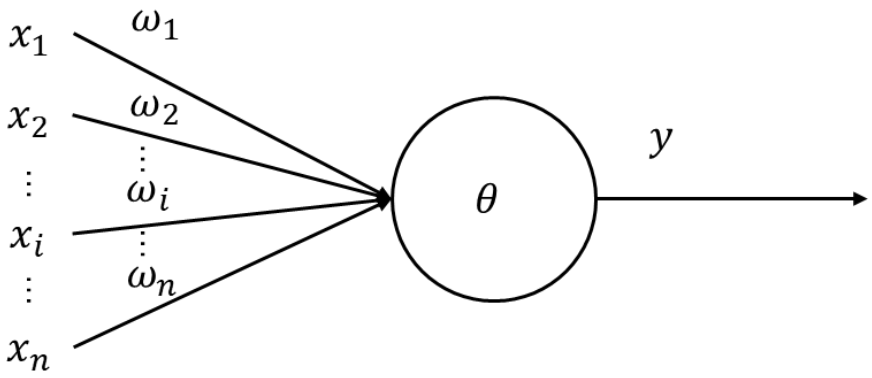


图 2.1 M-P 神经元模型

这个模型可以用下面的公式表示：

$$y = \theta(\sum_{i=1}^{n} \omega_i x_i + \omega_0) \tag{2-15}$$

也称作阈值逻辑单元(threshold logic unit, TLU)。其中，$\theta(\cdot)$为单位阶跃函数，如图所示。

感知机（Perception）由两层神经元组成。如图所示，输入层接收外部输入信号并将其传递给输出层，输出层是 M-P 神经元。感知功能可以轻松实现逻辑 AND、OR 和 NOT 操作。

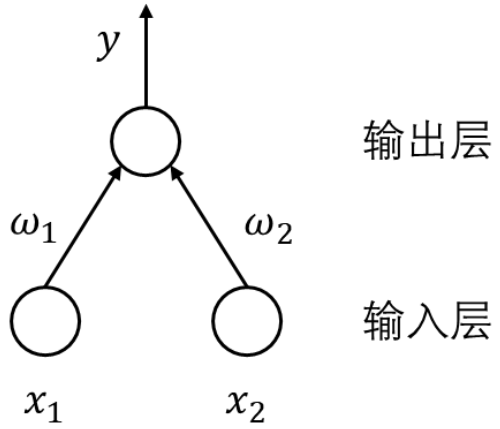


图 2.2 感知机模型

更一般地，给定训练集$\hat{y}$，可以学习权重和阈值。对于训练示例，如果当前感知器的输出为y，则感知器权重调整如下：

$$\omega_i \leftarrow \omega_i + \Delta\omega_i \tag{2-16}$$

$$\Delta\omega_i \leftarrow \eta(y - \hat{y})x_i \tag{2-17}$$

其中$\eta \in (0,1)$称为学习率(learning rate)。

然而，感知器仅具有输出层神经元来执行激活函数处理，其学习能力非常有限。它只能解决线性可分性问题，但是无法解决 XOR 等简单的问题。要解决 XOR 这种非线性可分性问题，需要使用更多层的神经元。如图 2.3 图所示，输出层和输入层之间的一层神经元称为隐藏层，隐含层和输出层神经元都具有激活功能的功能性神经元。

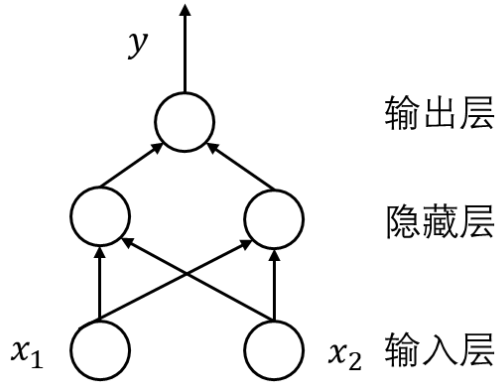


图 2.3 多层感知机结构

2.2.3 深度学习与反向传播算法

神经元越多，神经网络的参数越多，神经网络模型的复杂度也就越高，可以学习并解决更为复杂的问题。然而神经网络约复杂也意味着训练速度慢，以及过拟合等现象，因此很难被研究人员所青睐。随着大数据的发展以及计算机硬件能力的大幅提升，丰富

计算资源可以显著的提高神经网络的训练速度，大数据意味着更多的训练数据，大量的训练数据可以显著的降低过拟合问题发生的频率。作为复杂神经网络的代表，深度学习（deep learning）已经开始获得更多研究者的青睐。

典型的深度学习模型是深度神经网络。多层网络的学习能力远强于单层网络。然而，当训练多层网络时，上述公式（2-16）是不可行的，需要更好的学习算法，反向误差传播（error Back-Propagation，BP）算法是最优秀的代表之一。

给定训练集$D=\{(x_1,y_1),(x_2,y_2),\dots,(x_m,y_m)\}$，$x_i\in\mathbb{R}^d$，$y_i\in\mathbb{R}^l$，即输入示例由$d$个属性描述，输出$l$维实值向量。设一个拥有$d$个输入神经元、$l$个输出神经元。考虑第$n$层，其第$j$个神经元的阈值用$\theta_j$表示，第$n-1$层第$h$个神经元与第$n$层第$j$个神经元直接的连接权为$\omega_{hj}$。则第$n$层第$j$个神经元的输入为$\beta_j=\sum_{h=1}^{q}\omega_{hj}b_h$，其中$b_h$为$n-1$层第$h$个神经元的输出。设使用 Sigmoid 函数作为激活函数（图 2.4）。

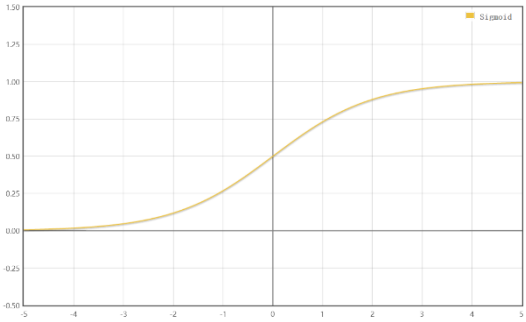


图 2.4 Sigmoid 函数图

对训练样例(x_k,y_k)，假定神经网络的输出为$\hat{y}_k=(\hat{y}_1^k,\hat{y}_2^k,\dots,\hat{y}_l^k)$，即

$$\hat{y}_j^k=f(\beta_j-\theta_j) \tag{2-18}$$

则网络在(x_k,y_k)上的均方误差为

$$E_k=\frac{1}{2}\sum_{j=1}^{l}(\hat{y}_j^k-y_j^k)^2 \tag{2-19}$$

BP 算法是一种经典的迭代算法、它采用梯度下降(gradient descent)的方法，即沿着误差的梯度反方向迭代更新参数，对于误差E_k，给定学习率η，有

$$\Delta\omega_{hj}=-\eta\frac{\partial E_k}{\partial\omega_{hj}} \tag{2-20}$$

$$\frac{\partial E_k}{\partial\omega_{hj}}=\frac{\partial E_k}{\partial\hat{y}_j^k}\cdot\frac{\partial\hat{y}_j^k}{\partial\beta_j}\cdot\frac{\partial\beta_j}{\partial\omega_{hj}} \tag{2-21}$$

根据β_j的定义，可以得到

$$\frac{\partial\beta_j}{\partial\omega_{hj}}=b_h \tag{2-22}$$

于是，根据公式（5.3）与（5.4）有

$$g_j = \frac{\partial E_k}{\partial \hat{y}_j^k} \cdot \frac{\partial \hat{y}_j^k}{\partial \beta_j} = \hat{y}_j^k(1-\hat{y}_j^k)(y_j^k - \hat{y}_j^k) \tag{2-23}$$

将式（5.10）和（5.8）代入，即 BP 算法中的ω_{hj}更新公式

$$\omega_{hj} = \eta g_j b_h \tag{2-24}$$

2.3 深度强化学习算法

2.3.1 深度强化学习发展现状

感知能力与决策能力是衡量人工智能的重要标准。然而，强化学习方法作为一种人工智能方法，在使用图像信息、语音信息等高维度的信息作为输入，直接进行决策甚至直接实现智能体的控制仍然是一个具有挑战性的问题。强化学习经过多年的发展，在理论上已经取得了很多突破，但是大多数成功的基于强化学习的智能体仍然依赖于人工提取特征，而且强化学习模型训练结果的好坏在很大程度上也取决于人工提取的特征品质。强化学习在处理感知问题时，存在着一定的劣势。深度学习方法更加适用于感知问题，它可以直接从图像、语音等高维数据中直接进行特征提取。因此，将两者结合起来，即将感知与决策相结合，两种方法可以相互弥补对方的不足，实现更高程度的人工智能，解决更为复杂的问题。

深度 Q 学习是深度强化学习(Deep Reinforcement Learning, DRL)领域的开创性方法，在最初的版本中，它使用雅达利游戏作为研究的对象，将游戏中每个连续四帧的画面作为算法的输入，通过深度卷积网络对原始图像进行特征提取，然后通过全连接的深度神经网络进行处理，最后输出游戏摇杆上每一个按键对应的 Q 值，实现了让智能体玩雅达利游戏的任务。

在深度 Q 学习方法提出之前，深度强化学习也存在很多研究。Shibata 等人提出了一种基于深度强化学习的移动机器人运动规划方法，它使用浅层神经网络对摄像机原始图像信号进行降维处理，将降维后的数据作为强化学习的输入，实现了移动机器人推箱子任务[66][67]。Lange 等人提出了一种基于深度强化学习的汽车运动控制方法，它使用深层的神经网络对视觉信号以及跑道信息进行输入降维，提取的低维特征作为强化学习的输入[68]，他将这种方法命名为深度拟合 Q 学习(deep fitted Q learning)。Koutnik 等人提出了一种基于深度强化学习的汽车自动驾驶方法，它首先使用神经进化(neural evolution, NE)方法对视觉信号进行降维处理，将降维后的信号作为强化学习模型的输入，实现了 TORCS 赛车游戏的自动驾驶任务[69]。

深度 Q 网络出现后，深层强化学习已成为先进人工智能的研究热点。深度学习将决策与感知，人工智能中最重要的两项能力同时进行了处理，大大提高了人工智能方法的可能性，深入其研究不仅是人工智能领域未来的发展前景，同时其强大的适应能力可以应用于非常多的领域，具有深远的研究意义与价值。

在机器人的运动规划领域，深度强化学习有着非常多的应用，Zhang 等提出了一种基于深度 Q 学习的运动规划方法，使用原始图像数据作为输入，可以完成机械手抵达目标位置的运动规划任务[70]。Gu 等提出了一种基于深度 Q 学习的运动规划方法，使用原始图像数据作为输入，可以完成机械手开门的运动规划任务[71]。Lei 等实现了基于深度 Q 学习的运动规划方法，使用原始图像数据作为输入，可以完成移动机器人躲避障碍并抵达目标的运动规划任务[72]。

2.3.2 深度强化学习基本框架

深度强化学习结合了强化学习的决策能力以及深度学习的感知能力，它可以根据传感器输入直接控制。它是一种更接近人类思维的人工智能方法。框架如图 2.5 所示。感知能力与决策能力是衡量人工智能的重要标准，可以认为深度学习（深度神经网络）是进一步增强感知能力并取得重大突破的核心技术。同时，强化学习的学习机制表明它不断与环境相互作用（可以看作是决策系统与环境之间的博弈），最优策略是通过反复试验获得的，这是决策的关键技术。

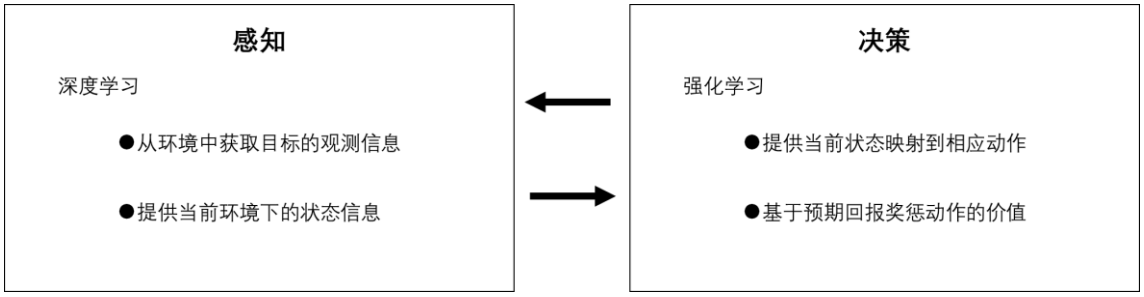


图 2.5 深度强化学习框架图

2.4 本章小结

本章对深度强化学习进行了介绍。首先介绍了强化学习理论，介绍了强化学习的发展现状，强化学习的基础马尔可夫决策模型，以及基于策略梯度的强化学习方法；然后介绍了神经网络理论原理，介绍了深层神经网络即深度学习的发展现状，还说明了神经网络的反向传播优化(BP)方法；最后介绍了深度强化学习理论的发展现状与其基本框架。

第3章 水下机器人未知环境运动规划模型

3.1 水下机器人模型与传感器模型

本课题所研究的载体模型是哈尔滨工程大学水下机器人技术重点实验室研发的某型号机器人，该机器人质量 98kg，长 2.01m，是一款小型智能水下机器人。该机器人可用于执行水文调查、管道检查、快速环境评估以及水下机器人科学研究等任务。该机器人主要搭载的传感器有 DVL 以及避障声纳，DVL 用于测速，避障声纳用于探测环境信息。机器人在尾部共有左右两个主推进器，可以在水平面上进行差速驱动。AUV 通过惯导系统进行水下导航。其外观如图 3.1 所示。

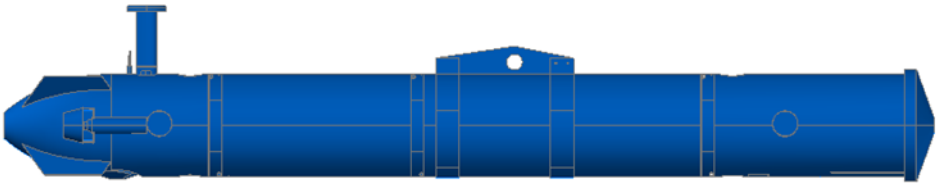


图 3.1 AUV 模型外观图

本文将基于上述机器人的模型，根据可以获得的传感器信息与执行机构的操纵性能等多约束目标，实现水下机器人的运动规划系统。在 AUV 的运动规划过程中，使用避障声呐传感器来检测机器人周围的障碍物信息。图 3.2 展示了 AUV 的避障声呐分布情况，一共存在 8 个避障声呐。定义 AUV 艏部的避障声纳为 1 号声纳，其余声纳按照顺时针方向依次定义为 2，3，4，5，6，7，8 号声呐，1、2、3 号声呐以及 1、7、8 号声呐之间夹角为45°；5 号声呐朝向 AUV 正后方，4、5、6 号声呐之间夹角为90°。避障声呐的波束宽度为17° ± 2°，可靠量程为1~20m。在本文中，不讨论传感器信息融合或滤波等问题，假设声呐传感器可以理想的探测到 8 个方向上的障碍物距离信息。

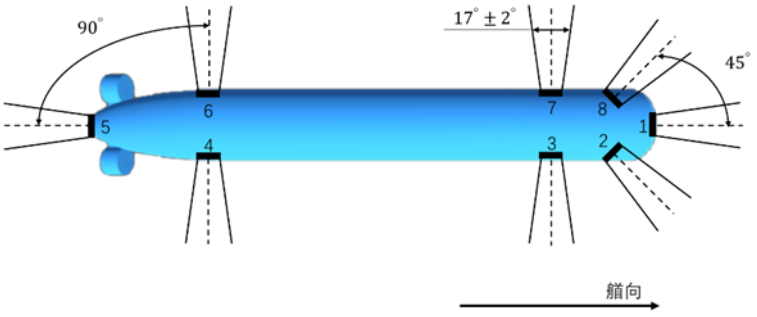


图 3.2 AUV 避障声呐模型图

AUV 配备 Link Quest 公司生产的 Nav Quest 600 Micro DVL 传感器，用于捕捉机器人在水下航行时的速度信息。Nav Quest 系列 DVL 具有远距离，高精度特性，可实现水下机器人的精确导航和定位，并可用于各种任务，包括 AUV 导航和 ROV 速度测量。可以测量AUV的纵向速度以及横向速度。该DVL工作频率为600Hz，精度$1\% \pm 1\mathrm{mm/s}$，量程为$\pm 20\mathrm{knots}$。

3.2 水下机器人的运动模型

3.2.1 水下机器人操纵性模型

水下机器人三自由度水平面操纵性模型基于刚体动力学方程[73]：

$$M_{RB}\dot{v} + C_{RB}(v)v = \tau_{RB} \tag{3-1}$$

其中

$$\tau_{RB} = \tau_{hyd} + \tau_{hs} + \tau_{wind} + \tau_{wave} + \tau \tag{3-2}$$

水平面$\tau_{hs} = 0$，并且假设$\dot{v}_c = 0$，可以得到：

$$\tau_{hyd} = -M_A\dot{v} - C_A(v_r)v_r - D(v_r)v_r \tag{3-3}$$

将公式（3-1）、（3-2）以及（3-3）组合，可以得到

$$\dot{\eta} = J_\Theta(\eta)v \tag{3-4}$$

$$M\dot{v} + C_{RB}(v)v + N(v_r)v_r = \tau + \tau_{wind} + \tau_{wave} \tag{3-5}$$

其中

$$N(v_r) \coloneqq C_A(v_r) + D(v_r) \tag{3-6}$$

此外，附加质量科里奥利力与向心力与粘性水动力一同组成了矩阵$N(v_r)$。这么作是因为很难区分$C_A(v_r)$与$D(v_r)$中的相关项。因此，为了避免过度参数化，模型中只使用了这些术语的总和。

在不考虑海流与风力的情况下，可以使用速度向量v替代公式（3-5）中的相对速度向量，以避免式中出现v_r，并且$\tau_{wave} = 0$，$\tau_{wave} = 0$。因此，公式（3-5）可以简化为：

$$M\dot{v} + C(v)v + N(v)v = \tau \tag{3-7}$$

其中

$$M = M_A + M_{RB} \tag{3-8}$$

$$C(v) = C_A(v) + C_{RB}(v) \tag{3-9}$$

在这个表示中，速度v是唯一的速度矢量，而不像公式（3-5）中既有v又有v_r。

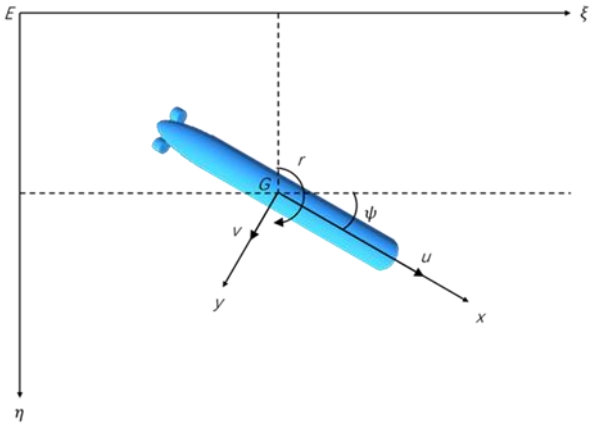


图 3.3 水下机器人运动坐标系图

水下机器人的水平面运动可以看作由纵荡、横荡以及转首三部分运动组成。状态可以由向量$v=[u,v,r]^T$和$\eta=[N,E,\psi]^T$表示，如图 3.3 所示。这就意味着，水下机器人的起伏、滚动和音调的运动被忽略了$(w=p=q=0)$。对于水下机器人的水平运动，运动方程从一般的六自由度表达式简化到只包括相对 z 轴的旋转运动。

$$J_\Theta(\eta)=R(\psi)=\begin{bmatrix}\cos(\psi) & -\sin(\psi) & 0\\ \sin(\psi) & \cos(\psi) & 0\\ 0 & 0 & 1\end{bmatrix} \tag{3-10}$$

同时本文假设，水下机器人具有均匀的质量分布并且关于xz平面对称，因此

$$I_{xy}=I_{yz}=0 \tag{3-11}$$

同样假设机器人左右对称，因此运动坐标系下y轴重心坐标为 0，即$y_g=0$。根据之前的假设，与刚体运动相关的矩阵简化为：

$$M_{RB}=\begin{bmatrix}m & 0 & 0\\ 0 & m & mx_g\\ 0 & mx_g & I_z\end{bmatrix} \tag{3-12}$$

$$C_{RB}(v)=\begin{bmatrix}0 & 0 & -m(x_g r+v)\\ 0 & 0 & mu\\ m(x_g r+v) & -mu & 0\end{bmatrix} \tag{3-13}$$

可以注意到，由于系统惯性矩阵的对称性，浪涌与摇摆和偏航是解耦的。可以得到机器人的附加质量相关矩阵为：

$$M_A=\begin{bmatrix}-X_{\dot u} & 0 & 0\\ 0 & -Y_{\dot v} & -Y_{\dot r}\\ 0 & -Y_{\dot r} & -N_{\dot r}\end{bmatrix} \tag{3-14}$$

$$C_A(v)=\begin{bmatrix}0 & 0 & Y_{\dot v}v+Y_{\dot r}r\\ 0 & 0 & -X_{\dot u}u\\ -Y_{\dot v}v-Y_{\dot r}r & X_{\dot u}u & 0\end{bmatrix} \tag{3-15}$$

其中$\mathrm{M}=M^T$，$C_{RB}(v)=-C_{RB}^T(v)$，$C_A(v)=-C_A(v)^T$，因此

$$M=\begin{bmatrix} m-X_{\dot{u}} & 0 & 0 \\ 0 & m-Y_{\dot{v}} & mx_g-Y_{\dot{r}} \\ 0 & mx_g-Y_{\dot{r}} & I_z-N_{\dot{r}} \end{bmatrix} \tag{3-16}$$

本文使用一个简化的非线性模型来描述$N(v)$，保留了最重要的操纵和推进损失。这个模型将交叉流阻力积分的二阶模量函数拟合

$$N(v)v=C_A(v)v+D(v)v \tag{3-17}$$

$$C_A(v)v=\begin{bmatrix} Y_{\dot{v}}vr+Y_{\dot{r}}r^2 \\ -X_{\dot{u}}ur \\ (X_{\dot{u}}-Y_{\dot{v}})uv-Y_{\dot{r}}ur \end{bmatrix} \tag{3-18}$$

$$D(v)v=\begin{bmatrix} -X_{|u|u}|u|u \\ -Y_{|v|v}|v|v-Y_{|v|r}|v|r-Y_{v|r|}v|r|-Y_{|r|r}|r|r| \\ -N_{|v|v}|v|v-N_{|v|r}|v|r-N_{v|r|}v|r|-N_{|r|r}|r|r| \end{bmatrix} \tag{3-19}$$

从上述等式中可以得到

$$C_A(v)=\begin{bmatrix} 0 & 0 & Y_{\dot{v}}v+Y_{\dot{r}}r \\ 0 & 0 & -X_{\dot{u}}u \\ -Y_{\dot{v}}v-Y_{\dot{r}}r & X_{\dot{u}}u & 0 \end{bmatrix} \tag{3-20}$$

$$D(v)=\begin{bmatrix} -X_{|u|u}|u| & 0 & 0 \\ 0 & -Y_{|v|v}|v|-Y_{|r|v}|r| & -Y_{|v|r}|v|-Y_{|r|r}|r| \\ 0 & -N_{|v|v}|v|-N_{|r|v}|r| & -N_{|v|r}|v|-N_{|r|r}|r| \end{bmatrix} \tag{3-21}$$

3.2.2 水动力系数

根据公式(3-22)、(3-18)、(3-19)、(3-20)以及(3-21)可以看出，在运动规划中考虑 AUV 的执行机构操纵性能时，需要已知 AUV 的水动力系数。水动力系数可以通过理论计算、近似计算以及实验测量等方法来获得。本文的研究中使用实验测量结合近似计算的方法来获得水动力系数。

$X_{|u|u}=-5.9\times10^{-3}$　　$Y_{|v|v}=-1.6687\times10^{-1}$　　$Y_{|v|r}=0$

$Y_{|r|r}=1.258\times10^{-2}$　　$N_{|r|r}=-1.2432\times10^{-1}$　　$N_{|v|r}=0$

$Y_{\dot{r}}=9.4196\times10^{-4}$　　$X_{\dot{u}}=-1.5777\times10^{-3}$　　$N_{|v|v}=0$

$Y_{\dot{v}}=-3.0753\times10^{-2}$　　$N_{\dot{r}}=-1.012\times10^{-1}$

3.3 基于多约束目标的运动规划问题建模

3.3.1 无地图未知环境运动规划模型

水下机器人的运动规划是一个复杂的多约束问题，其主要任务是在抵达目标点的同时，躲避障碍物。在无地图的未知环境下，无法预先规划轨迹，只能通过传感器信息来了解环境与自身状态，再输出规划策略，而且有着很高的实时性要求。根据操纵性方程

可以看出，需要得到水下机器人每一时刻获得的推力，来进行分析。在未知的环境下，需要直接将每一时刻传感器获得的信息映射成为机器人每一时刻的推进力。完整的推力输出序列可以使机器人在躲避障碍物的同时按顺序抵达目标点。为此可以设计一个数学模型来描述该问题：

$$\tau_t = f(s_t) \tag{3-23}$$

$$s_t = (x_t, p_t, v_{t-1}) \tag{3-24}$$

其中，s_t代表水下机器人在t时刻的状态与可以获得的环境信息。τ_t代表机器人在t时刻，机器人输出的纵向力与偏航力矩。s_t包括三个部分，其中x_t代表机器人通过避障声呐获得的障碍物信息；p_t代表目标点相对于水下机器人的位置信息；$v_{t-1} = [u, v, r] \in \mathbb{R}^3$代表在$t-1$时刻机器人的实际的速度信息。机器人通过避障声呐获得障碍物信息，通过其他传感器获得自身状态以及目标点位置信息，函数将这些信息映射为推进器推力，可以实现实时的“端到端”的运动规划。

考虑到对于强化学习方法，实现“端到端”的运动规划系统，即使用一个神经网络直接输出推力映射过于复杂，学习难度很高。本文采取了一种折中的学习方法，使用深度强化学习训练得到机器人的目标速度与目标艏向，再使用 S 面控制器进行控制的规划方法。

因此，水下机器人无地图环境运动规划问题模型可以转化为：

$$(u_t^g, \psi_t^g) = f(s_t) \tag{3-25}$$

$$s_t = (x_t, p_t, v_{t-1}) \tag{3-26}$$

输出的u_t^g代表 AUV 的目标速度，ψ_t^g代表 AUV 的目标艏向，将u_t^g与ψ_t^g传送给控制器，通过控制器完成整个控制系统。除此之外，本文所实现的运动规划问题遵循两个假设：

（1）水下机器人运动规划任务在水平面完成，拥有三个自由度。

（2）为了满足实时性的要求，控制系统与控制系统可以有规律的以 0.5s 的间隔进行控制输出，即每一次迭代间隔 0.5s。

3.3.2 状态空间

未知环境运动规划所需要的状态空间s_t包括机器人通过避障声呐获得的障碍物信息、目标点相对于水下机器人的位置信息以及机器人在上一时刻的实际的速度信息。

x_t代表障碍物的信息，本文假设机器人可以获取到环绕机器人一周 8 个方向障碍物信息，具体描述为 AUV 可以获取 8 个避障声呐方向上的障碍物距离信息，声呐模型如

图 3.2 所示。x_t是一个 8 维的状态。

p_t代表目标点相对于水下机器人的位置信息，以 AUV 重心为原点建立极坐标系，目标点所在的位置坐标即为p_t。p_t在机器人初始位置与目标点位置已知的情况下可以通过惯导系统获得。p_t是一个二维的状态。

v_{t-1}代表机器人在上一时刻的速度，在水下机器人操纵性方程中可以看出，AUV 规划的轨迹与自身速度有很大关联，将其作为状态空间的输入。v_{t-1}可以通过 DVL 结合惯导系统来获得。$v_{t-1} = [u, v, r] \in \mathbb{R}^3$，这是一个三维的状态。综上所述，本文描述的运动规划问题的状态空间是一个 13 维的状态空间。除此之外，本文所有的输入均经过标准化处理。

3.3.3 动作空间

由于本文研究的问题是水平面的运动规划问题，而且机器人推进器配置为差速驱动，因此 AUV 可以输出的外力只包括纵向力以及偏航力矩。控制器只可以对纵向速度以及目标艏向进行控制，因此动作空间只能输出纵向的速度以及转艏角速度，不能输出横向速度，即输出端$(u_t^g, \psi_t^g) \in \mathbb{R}^2$。综上所述，本文描述的运动规划问题的动作空间是一个 2 维的动作空间。

本文将输出的速度进行了一定的剪裁，将u_t^g的输出剪裁为(-0.2,1)，这是由于并不鼓励机器人作向后的运动，而且推进器向后的推力自身较小，其中 1 代表 AUV 的额定前进速度。将ψ_t^g剪裁为(-1,1)，-1 代表向左最大转艏角度，1 代表向右最大转艏角度，最大转艏角度为50°。

3.4 水下机器人操纵性试验

3.4.1 水平面直航仿真试验

本文对算法依托的水下机器人进行操纵性能测试，首先考虑水平面上 AUV 的直线航行能力，本文的研究不考虑海流环境的影响。首先设置两个推进器同时在 1V、2V、3V、4V 以及 5V 五个不同的电压指令下工作，测试 AUV 在不同的推进器电压下直航速度，**错误！未找到引用源。**展示了该仿真的结果，可以看到在电压较低时，AUV 的速度变化约平滑。在电压为 1V 时，AUV 在大约 120s 时抵达最大速度 0.4m/s。在电压为 5V 时，AUV 在大约 20s 时可以抵达最大速度 2.25m/s，该速度即最大前进速度。

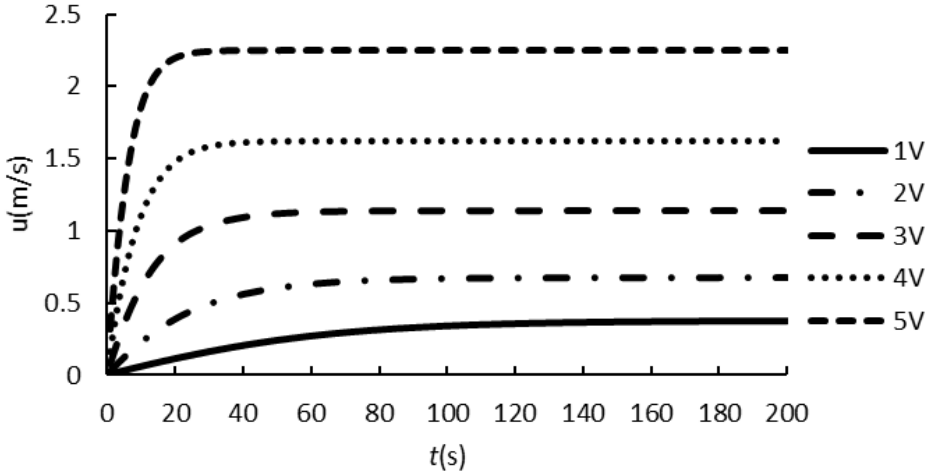


图 3.4 AUV 直航速度随时间变化图

3.4.2 水平面回转仿真试验

对 AUV 的回转运动能力进行仿真测试，回转运动能力的研究同样不考虑海流环境的影响。首先测试 AUV 在固定速度下的回转运动能力，保证 AUV 分别在 0.5m/s，1.0m/s 以及 1.5m/s 的情况下，进行调整推进器的电压，产生在该速度下最大的转艏力矩，图 3.6 展示了仿真试验效果。可以看出，在速度越大时，可以产生的转艏力矩越小，回转半径越大，当速度为 1.5m/s 时，回转半径约为 110m；在速度越小时，AUV 的回转操纵性能越好，回转半径较小，当速度为 0.5m/s 时，回转半径约为 60m。除此之外，当 AUV 的两个纵向推进器输出大小相同方向相反的推进力时，AUV 可以实现 0 航速下的原地旋转运动。

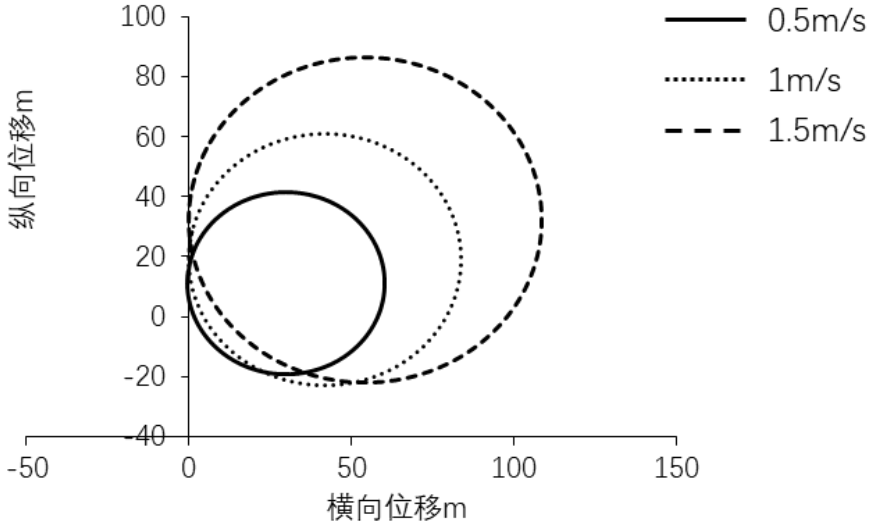


图 3.5 不同航速下 AUV 最大回转半径

除此之外，本文还测试了 AUV 在初始航速为 0 的情况下，分别直接给予推进器 0V-5V、1V-5V 以及 3V-5V 电压指令时，AUV 的回转操纵性能。图 3.6 展示了仿真试验的结果。从结果中可以看出在电压差值越大，转艏力矩越大，回转半径越小。在 0V-5V 的推进器电压指令情况下，AUV 的回转半径约为 20m，在电压差值为 4V 时，回转半径变化不大，但是在推进器电压指令为 3V-5V 的情况时，回转半径约为 60m。

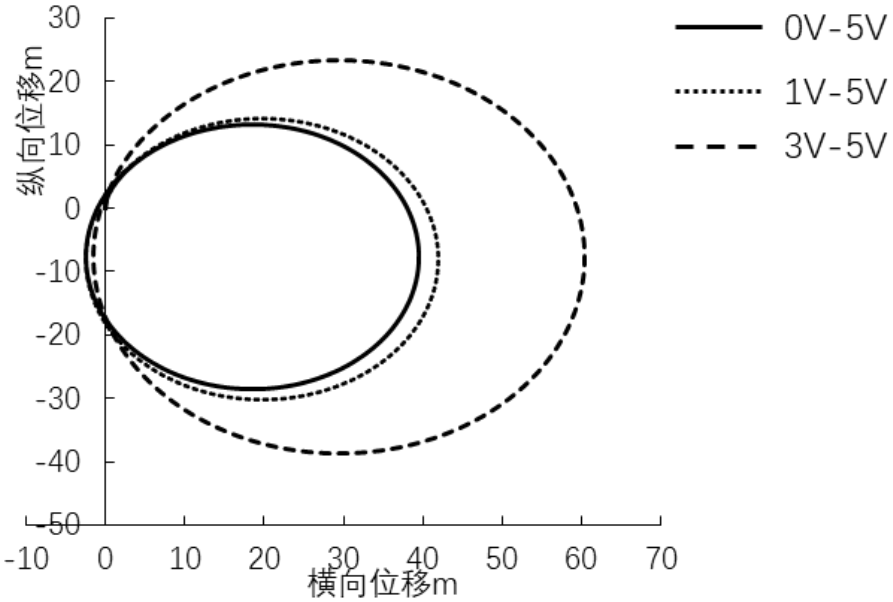


图 3.6 不同电压指令下 AUV 回转半径

3.5 水下机器人控制器

3.5.1 S 面控制方法

本文设计的运动规划系统输出为速度与目标转向角度，在仿真过程中还需要一个控制器。本文使用了基于 S 面控制方法的控制器。S 面控制器设计简单，控制效果良好。S 面控制方程为：

$$u = 2.0 \,/\, \left(1.0 + e^{(-k_1 e - k_2 \dot{e})}\right) - 1.0 \tag{3-27}$$

其中e和$\dot{e}$是控制器的输入，代表目标速度与目标角度的偏差值以及偏差变化率，需要进行标准化。u代表控制器的输出，在本文中是每个推进器的电压。k_1和k_2代表 S 面控制参数。

S 面控制需要对k_1和k_2两个控制参数进行调整，所需调整的参数相较于 PID 控制更为简单。S 面控制方法具有一定的劣势，例如 S 面控制方法在手动调整或者自适应调整两种调整方式下均无法实现最佳匹配。但是本文的主要内容是水下机器人的运动规划系

统，控制系统只是完成仿真规划的一部分，因此不对控制器进行更多的研究，而且由于 AUV 的复杂性，PID 控制方法也存在一定的近似空间。

3.5.2 控制仿真试验

本文对控制器进行了仿真试验，以测试控制器的性能。图 3.7 展示了控制器对于纵向速度的控制效果，设定目标速度为2.2m/s，速度控制效果较好，AUV 可以较为准确且平稳的达到目标速度。

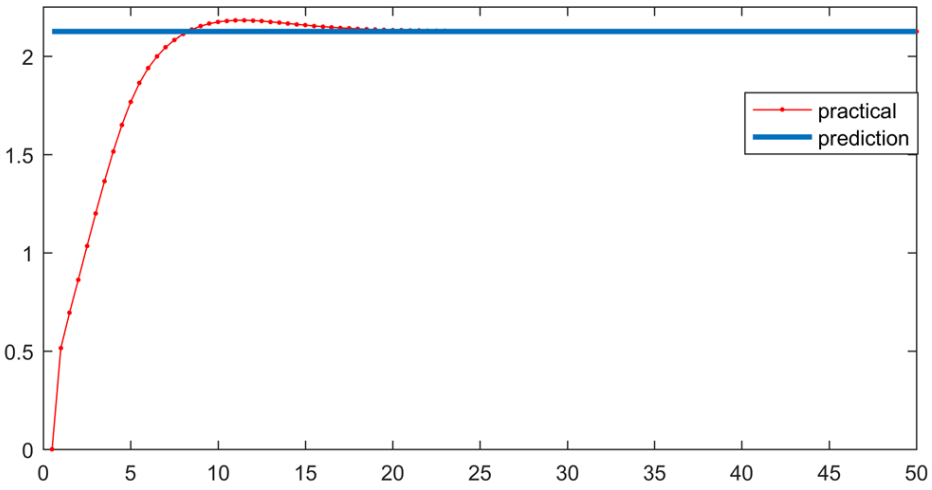


图 3.7 纵向速度控制效果

图 3.8 为艏向控制运动曲线效果，艏向目标位置为50°，初始艏向位置为 0，从曲线中可以看出，艏向控制效果较好，可以准确平稳地帮助机器人转至目标艏向。

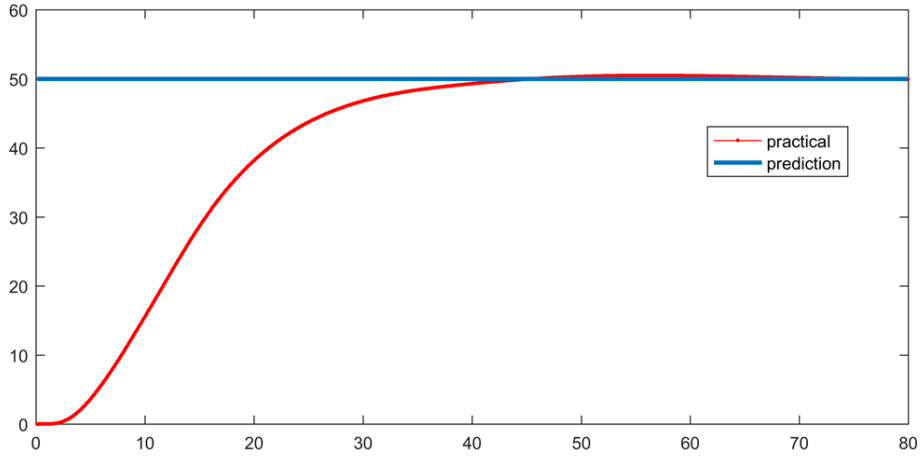


图 3.8 艏向控制效果

经过仿真试验，本文所使用的 S 面控制器控制效果良好，可以满足水下机器人运动规划系统的要求。

3.6 本章小结

本章建立了水下机器人的模型，阐述了机器人的传感器配置、具体参数以及声呐模型。分析并构建了水下机器人在无地图运动规划问题时的数学模型，将其与强化学习相结合，分析了深度强化学习输入的状态空间以及输出的动作空间。详细分析了水下机器人的操纵性方程。使用 S 面控制方法实现了机器人运动规划仿真平台所需要的控制器。这部分内容是本文研究的前提与基础。

第4章 基于深度强化学习的水下机器人未知环境运动规划方法

4.1 基于策略的深度强化学习算法

本文使用了基于策略梯度的深度强化学习方法。不同于基于值函数的强化学习方法通过判断不同动作的价值，选择价值最大的动作，基于策略梯度的方法直接寻找最优策略π。然而，基于值函数的深度强化学习方法，例如深度 Q 学习（DQN），在解决高维观测空间的问题时，只能同时处理离散和低维的动作空间。而在水下机器人运动规划任务时，需要具有连续的(实数值的)动作空间。DQN 不能直接应用于连续动作，因为它需要找到使动作值函数最大化的动作，这样在连续值情况下，需要的计算量非常大。

当然如果将动作空间离散化后，可以使用 DQN 方法。然而，这有许多限制，最大的问题是“维度灾难”：行为的数量会随着自由度的数量呈指数增长。AUV 共有六个自由度，即使每一个只进行非常简单的离散化即$a_i \in \{-k, 0, k\}$，会产生共$3^6 = 729$维度。如果想要对 AUV 进行更为复杂精细的规划，情况更为糟糕，因为需要在每一个自由度上进行更为精细的离散化，从而导致离散动作指数增长。过大的动作空间很难有效地探索，在这种情况下成功地训练 DQN 算法使非常困难的。除此之外，对动作空间的离散化会丢失关于机器人执行机构的约束信息，这些信息对于解决 AUV 运动规划问题是必不可少的。

设策略π_θ的目标函数为$J(\theta)$，则优化问题可以定义为：

$$\theta_{new} = \theta_{old} + \sigma \nabla_\theta J(\theta) \tag{4-1}$$

根据策略梯度定理，对于任何可微的策略π_θ，其策略梯度都可以表示为：

$$\nabla_\theta J(\theta) = \mathbb{E}_{\pi_\theta}[\nabla_\theta \, log \, \pi_\theta(a|s) \, Q^{\pi_\theta}(s,a)] \tag{4-2}$$

其中$Q^{\pi_\theta}(s,a)$代表行为价值，是对目标函数的评价函数，它可以是很多种函数，可以根据情况选择。

4.1.1 Actor-Critic 算法框架

本文使用 Actor-Critic（AC）框架来实现基于策略梯度的深度强化学习算法，AC 框架是一种求解基于策略的强化学习方法的有效框架[74][75][76]，框架如图所示。Actor-Critic 字面意思是演员与评论家，即演员根据环境执行动作，评论家对演员的动作进行评价，演员根据评价对执行动作的策略进行改进。具体描述为通过当前时刻的状态s_t，演员即机器人通过策略函数$\pi_\theta(a_t|s_t)$采样，得到控制器应该执行的动作a_t，探索环境，并根据

与环境的交互获得一个即时的奖励r_t，直到状态到达终止状态或t达到最大值。这个探索过程产生了一系列信息$\{(s_t, a_t, r_t), (s_{t+1}, a_{t+1}, r_{t+1}), \cdots\}$，评论家通过该信息对演员进行评价，鼓励高价值的行为，惩罚低价值的行为，来更新机器人的策略π_θ。评论家根据环境中采样轨迹中的奖励r来更新自身即行为价值函数$Q^{\pi_\theta}(s, a)$。

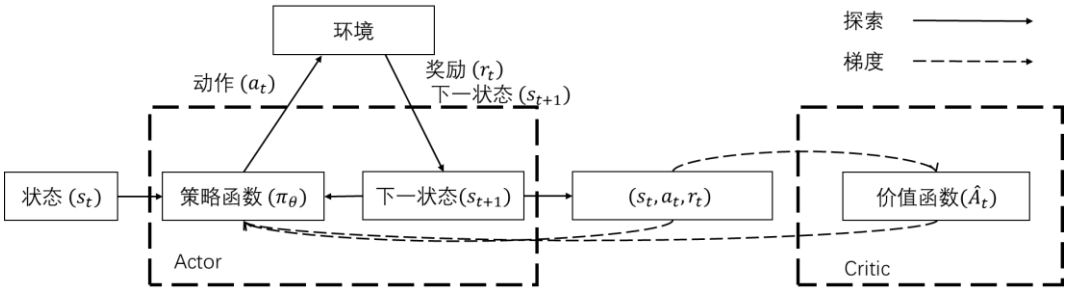


图 4.1 Actor-Critic 框架图

4.1.2 近端策略优化算法

策略的优化可以表示为公式（4-1），其中σ是一个超参数，用来控制沿梯度更新的速度，如果σ选择过小，策略的更新速度会十分缓慢，难以收敛，如果σ选择过大则有可能学习不好的策略，甚至导致算法崩溃。所以，合适的σ对于基于策略梯度的深度强化学习算法非常关键。

近端策略优化(Proximal Policy Optimization)算法是一种策略优化方法，它改进了普通的基于策略梯度算法无法保证策略无法保证策略性能单调非递减的问题[77]。

首先引入优势函数$A_\pi(s, a)$，根据公式（4-3）与（4-4）表示的状态价值函数与行为价值函数，优势函数可以表示为：

$$Q_\pi(s_t, a_t) = \mathbb{E}_{s_{t+1}, a_{t+1}, \cdots}\left[\sum_{l=0}^{\infty} \gamma^l r(s_{t+l})\right] \tag{4-3}$$

$$V_\pi(s_t) = \mathbb{E}_{a_t, s_{t+1}, \cdots}\left[\sum_{l=0}^{\infty} \gamma^l r(s_{t+l})\right] \tag{4-4}$$

$$A_\pi(s, a) = Q_\pi(s, a) - V_\pi(s), \text{where}$$
$$a_t \sim \pi(\cdot|s_t), s_{t+1} \sim P(s_{t+1}|s_t, a_t) \text{ for } t \geq 0 \tag{4-5}$$

使用$\hat{A}_t$作为$Q^{\pi_\theta}(s, a)$来表示行为价值，$\hat{A}_t$是优势函数的估计，使用$\hat{g}$表示策略梯度估计，$\hat{g}$可以表示为：

$$\hat{g} = \hat{\mathbb{E}}_t[\nabla_\theta \, log \, \pi_\theta(a_t|s_t) \, \hat{A}_t] \tag{4-6}$$

置信域策略优化(Trust Region Policy Optimization, TRPO)方法中提出[78]，目标函数$\hat{g}$可以在策略更新大小的约束下最大化，用公式表示为：

$$max_\theta \, \hat{\mathbb{E}}_t \left[\frac{\pi_\theta(a_t|s_t)}{\pi_{old}(a_t|s_t)} \hat{A}_t\right]$$

$$subject\ to\ \hat{\mathbb{E}}_t\left[KL\left[\pi_{\theta_{old}}(\cdot|s_t),\pi_\theta(\cdot|s_t)\right]\right]\le\delta \tag{4-7}$$

其中θ_{old}代表更新前的策略参数。设

$$r_t(\theta)=\frac{\pi_\theta(a_t|s_t)}{\pi_{old}(a_t|s_t)} \tag{4-8}$$

$r_t(\theta)$代表概率比，TRPO 优化的目标函数可以表示为：

$$L^{CPI}(\theta)=\hat{\mathbb{E}}_t\left[\frac{\pi_\theta(a_t|s_t)}{\pi_{old}(a_t|s_t)}\hat{A}_t\right]=\hat{\mathbb{E}}_t\left[r_t(\theta)\hat{A}_t\right] \tag{4-9}$$

上标 CPI 意味着保守的策略迭代(conservative policy iteration)[79]，如果没有约束，L^{CPI}的最大化将导致策略更新过大。因此可以在调整目标函数时，惩罚将策略改变过大即$r_t(\theta)$远离 1 的参数变化。即：

$$L^{CLIP}(\theta)=\hat{\mathbb{E}}_t\left[min\left(r_t(\theta)\hat{A}_t,clip(r_t(\theta),1-\varepsilon,1+\varepsilon)\hat{A}_t\right)\right] \tag{4-10}$$

其中$\varepsilon=0.2$是一个超参数，在min运算中，第一项$r_t(\theta)\hat{A}_t$为L^{CPI}，第二项$clip(r_t(\theta),1-\varepsilon,1+\varepsilon)\hat{A}_t$表示对策略概率比进行了剪裁，这一运算消除了不在区域$[1-\varepsilon,1+\varepsilon]$中的$r_t(\theta)$。最后，取裁剪了的目标函数和未裁剪目标函数的最小值，这样最终目标函数就是未裁剪目标的下界。

本文使用梯度法对$L^{CLIP}(\theta)$进行优化，其中$\hat{A}_t$具体可以表示为：

$$\hat{A}_t=\sum_{t'>t}\gamma^{t'-t}r_{t'}-V_\phi(s_t) \tag{4-11}$$

对 Critic 的优化即对$V_\phi(s_t)$的优化，其中ϕ代表 Critic 网络的参数，本文使用其方差$L_V(\phi)$作为 Critic 神经网络的损失函数，使用梯度下降法进行优化。其中$L_V(\phi)$公式为：

$$L_V(\phi)=\sum_{t=1}^{T}\left(\sum_{t'>t}\gamma^{t'-t}r_{t'}-V_\phi(s_t)\right)^2 \tag{4-12}$$

4.2 基于策略的深度强化学习算法的实现

4.2.1 激活函数

深度网络中激活函数的选择对训练过程和任务性能有很大影响。本文中使用的是 Swish 激活函数[80]，其函数表达式为：

$$f(x)=x\cdot sigmoid(x) \tag{4-13}$$

函数的图像如图 4.2 所示。

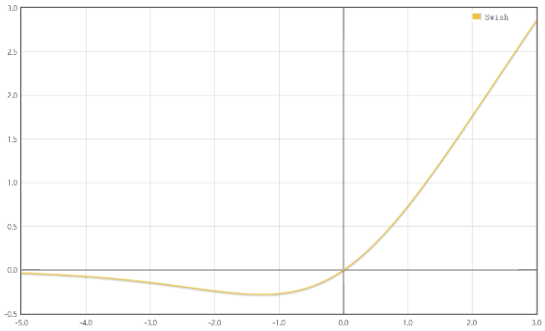


图 4.2 Swish 激活函数图

4.2.2 神经网络结构

本文使用的神经网络结构如图 4.3 所示。Actor 网络拥有三个全连接隐藏层，其中每个隐藏层包括 512 个节点，隐藏层之间使用 Swish 函数作为激活函数。使用 13 维的状态s_t作为网络输入，输出二维的控制目标$[u, \psi]$，该目标速度是一个归一化后的系数，将它乘以设定的最大目标速度来获得真实的目标速度。本文将输出的u剪切为$(-0.2,1)$，并不鼓励机器人作向后的运动。将输出的ψ剪切为$(-1,1)$。真实的目标速度维隐藏层与输出层之间使用线性激活函数。

Critic 网络同样拥有三个全连接隐藏层，其中每个隐藏层包括 512 个节点，隐藏层之间使用 Swish 函数作为激活函数。使用 13 维的状态s_t作为网络输入，输出为一个维度的价值$V_\phi(s_t)$。

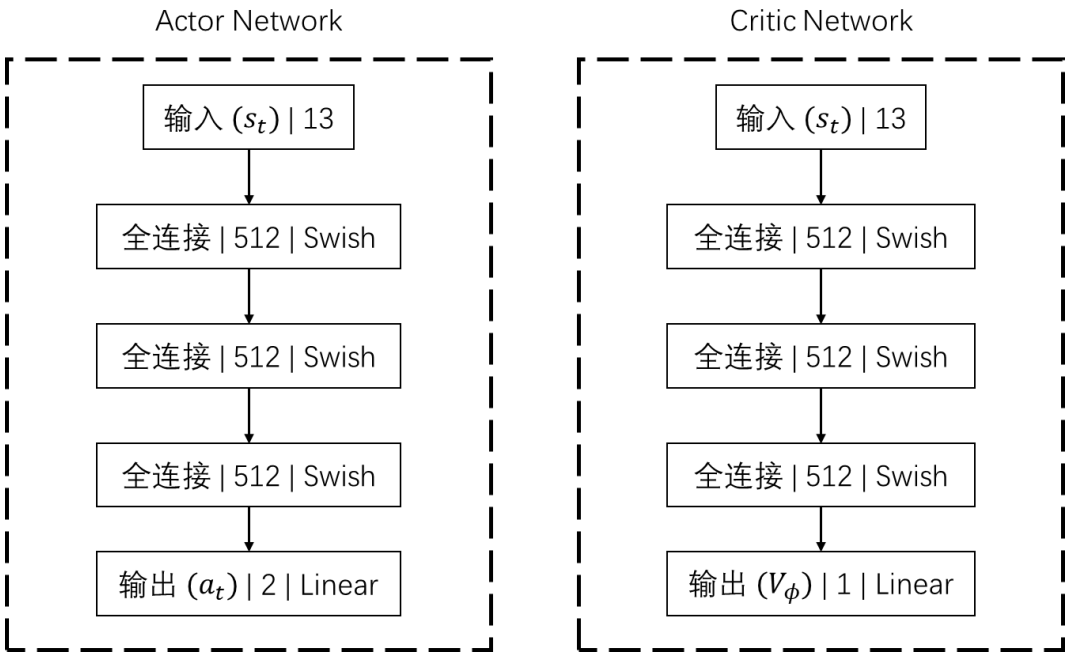


图 4.3 Actor 网络与 Critic 网络结构图

使用x^k代表神经网络任意第 k 层的输入数据，h^k代表第 k 层的输出，输入层为第 0 层，神经网络的权重与偏置分别为w^k和b^k，$activation^k$代表 k 层的激活函数，n^k表示第 k 层的节点数，则两种神经网络的计算过程均可以表示为：

$$h_i^k = activation^k\left(\sum_{i=1}^{n^{k-1}} x_i^k w_i^k + b^k\right) \tag{4-14}$$

4.2.3 奖励函数设计

奖励函数是指在强化学习算法的训练过程中，对智能体执行的动作优劣进行评价的标准，它在强化学习的训练过程中具有指引的功能。强化学习训练的目标是使智能体可以获得更多的行为价值。运动规划系统的目标是找到最优的策略逼近网络，帮助机器人在躲避障碍物的同时抵达目标点。因此，在设计奖励函数的过程中需要考虑运动规划策略的需求，优秀的奖励函数可以准确实时的评价当前的策略，以帮助训练。抵达目标点是运动规划的目标，因此应该对可以抵达目标点的策略进行鼓励。机器人的安全同样非常重要，因此需要对会发生碰撞的策略进行惩罚。运动规划所需时间越短，规划效果越好，因此应该鼓励积极抵达目标点的策略。因此，本文设计了的奖励函数如下所示：

$$r(s_t, a_t) = \begin{cases} r_a & if\ arrive \\ r_b & if\ collide \\ -1/5000 & every\ step \end{cases} \tag{4-15}$$

如果使用碰撞检测系统检测到模型抵达目标点，则模型会得到正奖励r_a，如果与障碍物放生碰撞，获得负奖励r_b，这两种情况均会使当前训练回合结束。每一步均会获得 -1/5000 的负奖励，以鼓励机器人快速寻找目标点。

4.2.4 算法流程

综上所述，本文提出的基于策略梯度深度强化学习的水下机器人未知环境运动规划算法如算法 1 所示。

算法 1 水下机器人未知环境运动规划算法

1：初始化 Actor 网络与 Critic 网络参数θ以及ϕ；
2：初始化训练环境；
3：AUV 根据策略$\pi_\theta(a_t|s_t)$以及状态s_t选择动作a_t；
4：观察执行a_t后 AUV 的状态s_{t+1}，并根据s_{t+1}与奖励函数得到奖励r_t；
5：收集数据集(s_t, a_t, r_t)并将其保存在内存D中；
6：判断s_{t+1}是否为终止状态，如果是，进行 8，不是则返回 3；
7：判断当前训练回合是否达到规定的最大步数t_{max}，如果是，进行 8，不是则返回 3；

8：使用公式（4-11）计算优势函数；
9：通过公式（4-10），使用梯度下降法优化 Actor 策略网络参数θ；
10：通过公式（4-12），使用梯度下降法优化 Critic 评价网络参数ϕ；
11：更新策略$\pi_{old} \leftarrow \pi_{\theta}$；
12：判断当前训练总步数是否达到训练要求的步数，如果是，进行 13，不是则返回 2；
13：算法结束。

4.3 课程学习

4.3.1 课程学习背景以及研究现状

课程学习[81]近年来在强化学习领域越来越受到关注。这种学习模式的灵感来自于人类和动物认知过程的学习原理，这些学习原理通常从学习任务中较容易的方面开始，然后逐渐考虑到更复杂的例子。这可以类比于人类教育的方式。在人类教育中，学生在学习更高级的代数课题之前，应该先理解初级代数。经验证明，这种学习模式有助于避免糟糕的局部极小值，并获得更好的泛化结果[82][83]。

课程由人为指定，且一直保持固定，它是一个按照学习难度升序排列的参数列表。课程学习可以灵活的结合来自不同来源的先验知识，加快训练的速度。

Bengio 等人提出了一种新的学习范式，称为课程学习(curriculum learning, CL)，通过在训练中逐渐包含从易到复杂的样本来学习模型，以增加训练样本的熵[81]。随后，Bengio 等人对这种学习范式的合理性进行了深刻的探索，讨论了课程学习与传统优化技术之间的关系[84][85]。从人类行为的角度来看，有证据表明课程学习符合人类教学的原则[82][83]。

课程通常是通过特定问题的启发式推理来获得的。例如，在几何形状分类的任务中，课程是由形状的变异性得到的，表现出较少变异性的形状应该更早学会进行分类[81]。在机器手抓取物体的运动规划任务中，开发者需要对物体是否可以抓起，抓起难度等内容进行排序，即设计课程[82]。在自然语言处理的任务中，问题可能的解会随着语言的长度呈指数增长，短句难度更低，应该在更早的课程中进行学习[86]。在智能体学习游玩游戏的任务中，难度过高的敌人可能导致策略无法学习到任何有效的策略，因此更应该使用游戏难度较低的敌人进行初期的学习[87]。

4.3.2 课程设计

本文的课程学习是基于训练环境的改变而实现的，课程可以分为两个部分，第一课

首先训练水下机器人寻找目标点的能力，第二课再训练其躲避障碍物的能力。具体细节如下：

第一课的内容为机器人抵达目标点。训练环境是一个无边界的二维环境，浅色的圆形代表目标点，深色的胶囊形代表机器人。目标的初始位置固定，机器人的初始位置以一定规则随机指定，规则如下：以目标点圆心为原点坐标建立极坐标系，那么机器人重心的初始位置在$(10, \psi_{auv})$处，其中$\psi_{auv} \in [0,2\pi]$是一个随机数，这意味着机器人总是出现在以目标点圆心为圆心，半径为$10m$的圆的边界上。以机器人重心为坐标原点建立极坐标系，那么目标点圆心的坐标$(10, \psi_{goal})$，其中$\psi_{goal} \in [0, \pi/4]$也是一个随机数，这意味着机器人的初始朝向与目标点圆心相对机器人重心的方向间的夹角不会超过45°，这样设计的目的是因为欠驱动水下机器人转向能力弱。课程的总目标是让机器人可以很快的找到目标点。如前文所述，本文使用碰撞检测系统进行检测机器人是否抵达目标点。机器人的声呐无法发现目标点。

图 4.4 课程 1 与课程 2 训练环境示意图

第二课的内容为机器人躲避障碍物，训练环境如图所示。训练环境与第一课类似，但是增加了数个深色的圆形以及矩形，它们代表障碍物。同样使用碰撞检测系统检测机器人是否碰到了障碍物，机器人的声呐可以发现障碍物。在第二课中，机器人与障碍物的初始位置固定，机器人初始朝向固定，目标点以一定规则随机出现，规则如下：以机器人重心为极坐标原点，那么目标点圆心的初始位置为$(100, \psi_{goal})$，其中$\psi_{goal} \in [0,2\pi]$是一个随机数，这意味着目标点总是出现在以机器人圆心为圆心，半径为$100m$的圆的边界上。课程的总目标是让机器人可以找到更远的目标点，同时训练其躲开障碍物的能力。

4.3.3 课程训练结果

本文使用 TensorFlow 进行的神经网络搭建，使用 Adam 优化器进行优化，使用 Titan V 显示处理器搭配 i9 7980XE 处理器，训练时间 20 小时，训练的具体参数如表 4.1 所示。

表 4.1 训练参数

参数	值
学习率	0.0003
γ	0.99
η	0.01
课程 1 训练步数	100000
课程 2 训练步数	900000
r_a	1
r_b	-1
t_{max}	200

模型经过课程 1 与课程 2 的完整训练后，在课程 2 训练环境中规划的轨迹如图 4.5 所示。与在训练时相同，机器人位于环境中心，目标点随机选取。AUV 在规划开始时先转向并朝着目标点移动，在声呐发现障碍物后向左转向，躲避开障碍物后再向目标点前进并抵达目标点，可以看出规划的轨迹较为平滑。对规划的轨迹进行定量分析，模型规划需要 97 步，规划时间为 48.68s，规划的路径总长度为 104.57m。

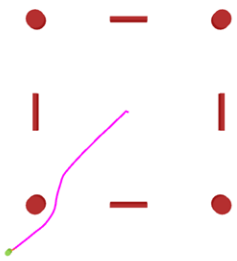
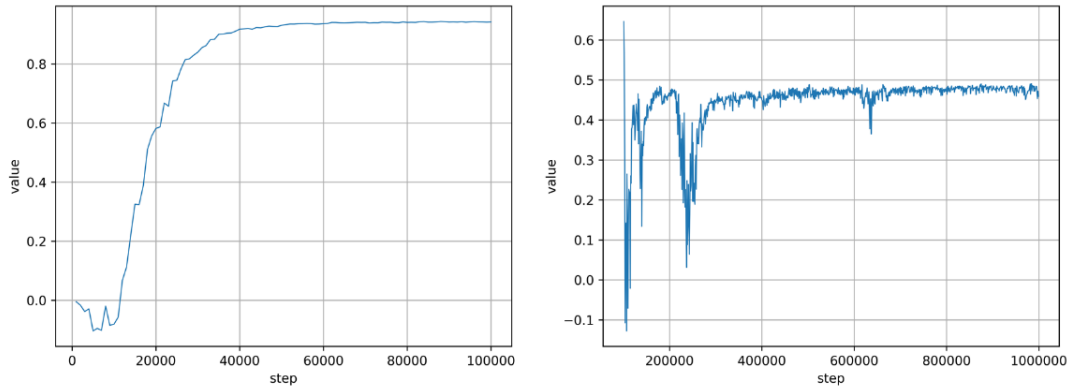


图 4.5 模型在课程 2 环境中规划轨迹图

图 4.6 中为训练过程中 AUV 每一回合可以获得的平均价值随着训练进行的变化曲线。其中图 4.6 (a)为课程 1 训练过程中的奖励变化，图 4.6 (b)为课程 2 训练过程中的变化，图中的横坐标训练步数为总计训练步数。可以看出，机器人在课程 1 的训练过程中，每一回合可以获得的奖励在一开始小于 0。在 10000 步训练到 30000 步之间，奖励迅速升高，而后逐渐趋向于收敛。这意味着模型已经找到了抵达目标点的策略。在课程 2 的训练过程中，机器人获得的奖励在一开始迅速下降，这是因为环境中存在障碍物，而机

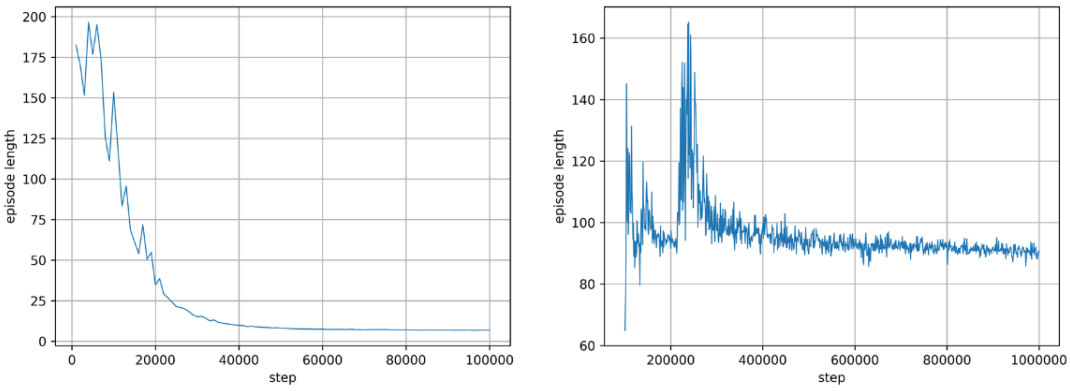
器人的策略还不能顺利躲避障碍物，在 400000 步训练前，奖励都十分波动，在此之后，机器人找到了可以顺利躲避障碍物的策略，因此奖励收敛。



(a) 课程1训练价值　　(b) 课程2训练价值

图 4.6 训练回合价值均值变化图

图 4.7 为模型在训练过程中，每次规划回合所需要的规划步数随着训练进行的变化图。其中图 4.7 (a)为课程 1 训练过程，图 4.7 (b)为课程 2 训练过程。同样可以看出，在课程 1 训练过程中，训练开始时，机器人无法抵达目标点，每一次训练都接近达到t_{max}，随着训练的进行步数逐渐减少并且收敛，在课程 1 中只有抵达目标点可以提前结束训练回合，因此这说明机器人可以越来越快的抵达目标点。在课程 2 训练过程中，初始步数非常不稳定，机器人有可能抵达目标点可有课程碰到障碍物。随着训练进行，步数逐渐增加，机器人开始尝试躲避障碍物，训练步数到达 400000 步时，机器人已经找到了可以躲避障碍物并且抵达目标点的有效策略，回合步数也趋于收敛。



(a) 课程1回合步数　　(b) 课程2回合步数

图 4.7 每回合规划步数变化图

本文对基于课程学习训练的深度强化学习模型与普通训练方法训练的深度强化学

习模型进行了对比。基于普通训练方法训练的模型使用了相同的框架、优化方法、神经网络结构以及训练参数。图 4.8 为两种训练模型的奖励图，其中课程训练模型为课程 1 与 2 组合在一起的奖励变化。从图中可以看出，基于课程学习训练的模型在训练速度上有着明显的优势。不使用课程学习训练的模型在 100000 步训练步数结束时，奖励也没有明显的增长，这意味着它并没有找到有效的策略。

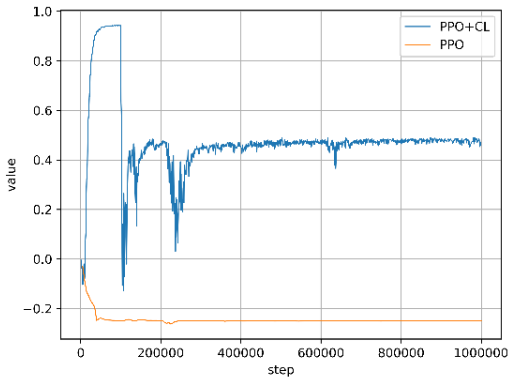


图 4.8 课程学习与无课程学习价值变化对比图

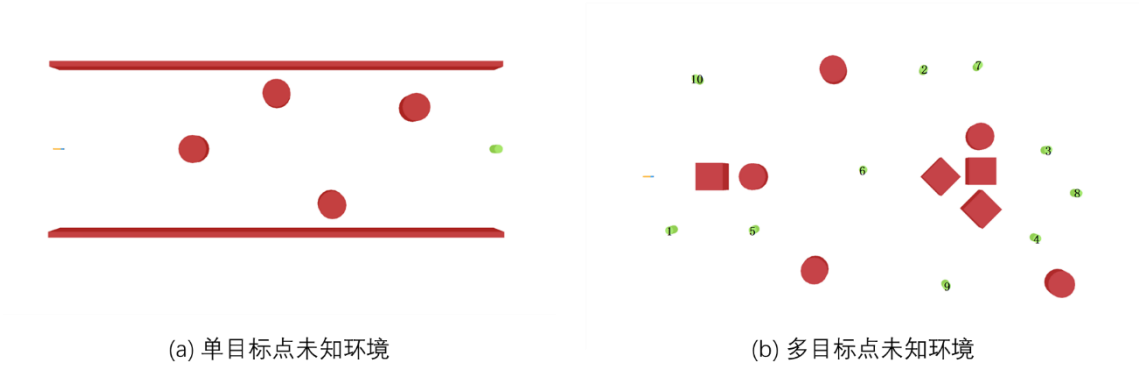
上述仿真试验结果表明，使用课程学习的训练方法训练的深度强化学习模型是收敛的，并且相比于普通训练方法，它有着更快的训练速度。并且模型在训练环境中可以顺利的完成运动规划任务，在躲避障碍物的同时抵达目标点，规划的轨迹也较为平滑。

4.4 未知环境仿真试验与结果分析

4.4.1 未知环境仿真实验设计

本文实现的基于策略的深度强化学习算法在未知环境下同样有着良好的适应能力。为了验证这一点，在未知环境下对训练好的算法模型进行仿真试验。本文设计了两种未知环境模型，分别为单目标未知环境以及多目标点未知环境。两种环境模型如图 4.9 所示。

其中图 4.9 (a)展示了单目标未知环境的模型，试验环境是一个无边界的二维环境，浅色的圆形代表目标点，深色的胶囊形代表 AUV，深色的圆形以及矩形代表障碍物。AUV 初始位于环境左侧，目标点初始位于环境右侧，其余条件均与训练时一致。图 4.9 (b)展示了多目标点未知环境的模型，环境图示同上。环境中共有 10 个目标点，图中对其进行了编号，AUV 需要按照编号的顺序依次抵达目标点，并躲避障碍物。AUV 初始位置位于环境左侧。

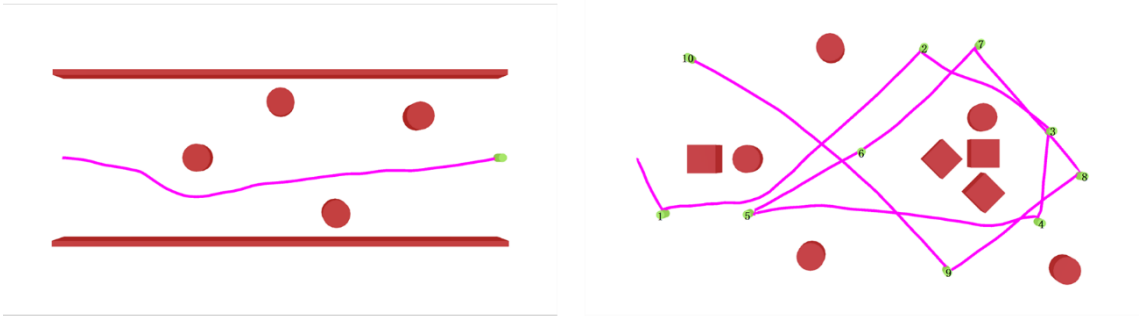


(a) 单目标点未知环境　　(b) 多目标点未知环境

图 4.9 单目标点未知环境模型示意图

4.4.2 未知环境仿真试验结果

本文使用两种未知环境对使用课程训练的基于策略的深度强化学习模型进行了仿真试验。图 4.10 中为模型规划的轨迹图。对规划的轨迹进行定性的分析，图 4.10 (a)为单目标未知环境的规划轨迹，可以看出，机器人顺利的躲开了障碍物并抵达了目标点，而且轨迹较为平滑。对规划的轨迹进行定量的分析，本文使用三个指标对规划的轨迹进行分析，分别为规划所需要的步数；规划所需要的时间；以及规划轨迹的距离。其中基于本文提出的方法训练的模型在单目标未知环境中规划的轨迹距离为 163.06m；规划的总步数为 143 步；规划的总时间为 71.52s。

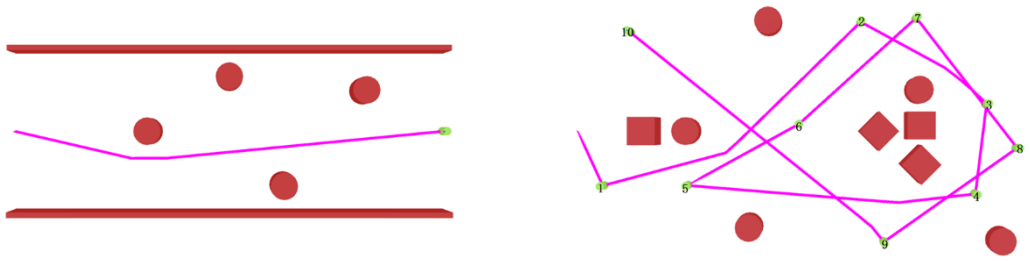


(a) 单目标点未知环境　　(b) 多目标点未知环境

图 4.10 课程学习模型未知环境规划轨迹示意图

图 4.10 (b)为模型在多目标未知环境中规划的轨迹图。对轨迹进行定性分析，可以看出机器人可以顺利的完成运动规划任务，但是规划并不平滑。对轨迹进行定量分析，基于本文提出的方法训练的模型在多目标未知环境中规划的轨迹距离为 687.70m；规划的总步数为 648 步；规划的总时间为 324.34s。在多目标的未知环境中，模型在规划时习惯于减速至停止，再转向寻找目标点，而不是在减速的同时调整方向。这是因为再训练过程中，模型倾向于让 AUV 在一开始调整好方向再前进，模型陷入了局部最优解。本文将在接下来的章节中研究这个问题。

除此之外，对本文提出的模型规划的轨迹与基于 A*算法规划的路径进行了对比。A*是一种在已知地图情况下的经典的全局规划算法，本文将单目标环境与多目标环境的地图直接作为 A*算法的输入，输出在地图已知情况下的规划路径。规划的路径如图 4.11 所示。



(a) 单目标点环境　　(b) 多目标点环境

图 4.11 A*算法规划轨迹示意图

从图 4.11 中可以看出，使用本文提出的模型规划的路径与已知地图情况下规划的路径相似。但是本文提出的模型所规划的路径也仍存在着一些问题，存在一些并不需要的转向动作。A*算法在单目标环境下规划的轨迹距离为 158.413m，在多目标环境下规划的轨迹距离为 676.50m。本文的模型相比于 A*的轨迹在单目标环境中距离多 2.9%，在单目标环境中距离多 1.7%，这意味着本文训练的模型所规划的轨迹已经相当接近于已知地图情况下的轨迹。本文的研究目标是未知环境下的轨迹规划，在已知地图的情况下基于图搜索的方法仍然是更好的解决办法。

4.5 本章小结

本章首先分析了基于策略的深度强化学习算法相比基于值函数的深度强化学习算法的优点，并阐述了本文所使用的 Actor-Critic 算法框架，与近端策略优化算法。然后介绍了本文的算法模型的具体实现方式，使用 Swish 函数作为神经网络的激活函数，Actor 神经网络与 Critic 神经网络的构建以及奖励函数的设计。使用课程训练的方法对算法模型进行训练，首先设计了适合于 AUV 运动规划的课程，与不使用课程训练的模型进行了对比，该模型可以更快的收敛，在训练环境中有着很好的规划效果。最后在未知环境中对该算法模型进行了仿真试验，AUV 可以顺利完成运动规划任务，证明了本文提出的使用课程学习训练的基于策略的深度强化学习算法模型对未知环境有着良好的适应能力。

第5章 基于内在好奇心模型的深度强化学习方法

5.1 “好奇心”奖励研究背景

强化学习算法的目标是通过最大化环境提供的奖励来实现目标任务的学习策略。在某些任务的训练过程中，环境可以持续地给智能体提供奖励，例如 Atari 游戏任务中的游戏得分[86]；在控制机械手的运动规划任务中，手臂与目标物体之间的距离[87]。然而，在许多的任务中，智能体可以获得的奖励非常缺少甚至完全缺失，无法构造一个有效的奖励函数。在水下机器人的运动规划任务中，只有当代理成功到达指定的目标状态时，智能体才可以获得积极的奖励以改进策略。如上文的例子，除了最简单的环境外，在复杂的环境中智能体偶然寻找到目标状态（即随即探索）的可能性微乎其微，智能体几乎无法获得外部的奖励。

当外部的奖励很少时，内在的奖励(intrinsic reward)或者说“好奇心”奖励就变得至关重要[88]。这种内部的奖励大多数情况下可以被分为两大类：1）鼓励智能体探索“新颖”的状态[89][90][91]；2）鼓励智能体执行可以减少它在预测自己的动作后果（下一状态）时的误差以及不确定性的动作[92][93][94][95][96]。

测量状态的“新颖”的程度需要建立一个环境状态分布的统计模型，测量预测误差以及不确定性需要建立一个环境的动态模型，这个模型根据当前的状态s_t以及t时刻执行的动作a_t来预测环境的下一状态s_{t+1}。智能体根据预测下一状态的难度，生成一个“好奇心”奖励。

考虑到海洋环境的复杂性以及噪声的干扰，只考虑环境中的由于智能体动作影响的状态变化。也就是说，本文对原始的状态空间进行了特征提取，转化称为另一个特征空间，这个特征空间只表示了与智能体执行动作有关的状态信息。本文使用一个神经网络来学习这个状态空间，然后用这个特征空间对动作进行反向预测。因为是使用特征提取后的状态特征空间对动作进行的反向预测，所以该特征空间不会嵌入不影响智能体的环境状态。

然后，本文使用这个特征提取后的状态特征空间训练一个正向模型(Forward Model)，该模型输入当前状态s_t以及t时刻执行的动作a_t，输出预测的下一状态s_{t+1}。而正向模型的预测误差则作为“好奇心”奖励（即内在奖励，intrinsic reward）提供给智能体以帮助其学习策略。

“好奇心”在解决奖励稀疏的任务中已经被广泛的研究。好奇心是学习的一种机制，

在迁移至未知环境中时理论上会有帮助，本文评估了基于“好奇心”奖励训练的模型在迁移到未知环境时的有效性。

5.2 基于内在好奇心模型的深度强化学习方法

本文的深度强化学习方法由两个子模型组成：1）输出内在奖励信号的“好奇心”模型；2）输出一系列规划动作以最大化奖励信号的策略模型。除了内在奖励外，AUV 还可以从环境中获得一些外部奖励。设 AUV 在t时刻产生的内在好奇心奖励为r_t^i，外部奖励为r_t^e。策略模型以最大化两个奖励的总和$r_t = r_t^i + r_t^e$为目标进行优化，其中r_t^e在大部分情况下为零。

5.2.1 策略模型

定义策略为$\pi_{\theta_P}(a_t|s_t)$，其中θ_P代表深度神经网络参数，给定智能体所在的状态s_t，策略模型根据策略进行采样动作，即动作$a_t \sim \pi_{\theta_P}(a_t|s_t)$，$\theta_P$需要被优化到使两个奖励总和最大，表示为：

$$\max_{\theta_P} \mathbb{E}\left[\nabla_\theta \, log \, \pi_{\theta_P}(a_t|s_t)\, \hat{A}_t\right] \tag{5-1}$$

除非特别说明，下文中使用符号$\pi(a|s)$来表示参数化的策略$\pi_{\theta_P}(a_t|s_t)$。

5.2.2 内在“好奇心”模型

内在“好奇心”模型由两个子模型组成：1）特征提取子模型；2）状态预测子模型。其中特征提取子模型对原始状态空间进行特征提取，剔除掉与 AUV 动作无关的状态，以消除噪声干扰，生成特征状态空间。状态预测子模型通过对下一特征状态空间进行预测，分析预测错误以及不确定性，生成内在奖励，鼓励 AUV 对不确定的状态进行探索。

本文使用了神经网络提取衡量好奇心时所使用的特征状态空间，帮助“好奇心”模型提供一个良好的、可以抵抗噪声干扰的内在奖励。为了学习得到优秀的特征提取模块，本文使用了两个神经网络来进行训练学习：1）第一个神经网络将原始状态s_t转化为一个特征向量$\phi(s_t)$；2）第二个神经网络输入$\phi(s_t)$与$\phi(s_{t+1})$连续两个状态特征提取后的特征向量，反向预测智能体从状态s_t移动到状态s_{t+1}所采取的动作a_t。特征提取模块的数学模型用g来表示，可以定义为：

$$\hat{a}_t = g(s_t, s_{t+1}; \theta_I) \tag{5-2}$$

其中$\hat{a}_t$代表对动作a_t的反向预测值，θ_I代表神经网络的参数，需要对其进行优化，优化的目标函数为：

$$\min_{\theta_I} L_I(\hat{a}_t, a_t) \tag{5-3}$$

其中L_I代表损失函数，度量预测动作$\hat{a}_t$与实际动作a_t之间的差异。这一部分模块也可以称之为反向模型(Inverse Model)，反向模型训练所需要的数据(s_t, a_t, s_{t+1})通过智能体使用$\pi(s)$与环境进行交互来获得。

本文使用一个神经网络来训练学习状态预测模块，输入a_t与$\phi(s_t)$，输出$t+1$时刻的特征状态空间的预测：

$$\hat{\phi}(s_{t+1}) = f(\phi(s_t), a_t; \theta_F) \tag{5-4}$$

其中$\hat{\phi}(s_{t+1})$代表对$\phi(s_{t+1})$的预测，θ_F代表神经网络的参数，通过最小化损失函数L_F进行优化

$$L_F\left(\phi(s_t), \hat{\phi}(s_{t+1})\right) = \frac{1}{2}\left\|\hat{\phi}(s_{t+1}) - \phi(s_{t+1})\right\|_2^2 \tag{5-5}$$

这个学习模型f也可以称之为正向模型(Forward Model)。

最终的内在奖励r_t^i通过以下公式计算：

$$r_t^i = \frac{\eta}{2}\left\|\hat{\phi}(s_{t+1}) - \phi(s_{t+1})\right\|_2^2 \tag{5-6}$$

其中$\eta > 0$是一个超参数，用来控制内在奖励的大小。为了产生基于“好奇心”的内在奖励信号，对公式（5-3）和公式（5-5）中所描述的正向模型损失和反向模型损失进行联合优化。反向模型通过训练，实现特征状态空间的提取，正向模型则通过该特征状态空间进行状态预测。该好奇心结构可以成为内在好奇心模块(Intrinsic Curiosity Module，ICM)。用以预测的特征状态空间不会受到不受 AUV 动作影响的环境状态的影响，智能体不会因为抵达本质上不可预测的状态而获得内在奖励，学习到的探索策略对环境噪声具有鲁棒性。其完整的结构如图 5.1 所示。

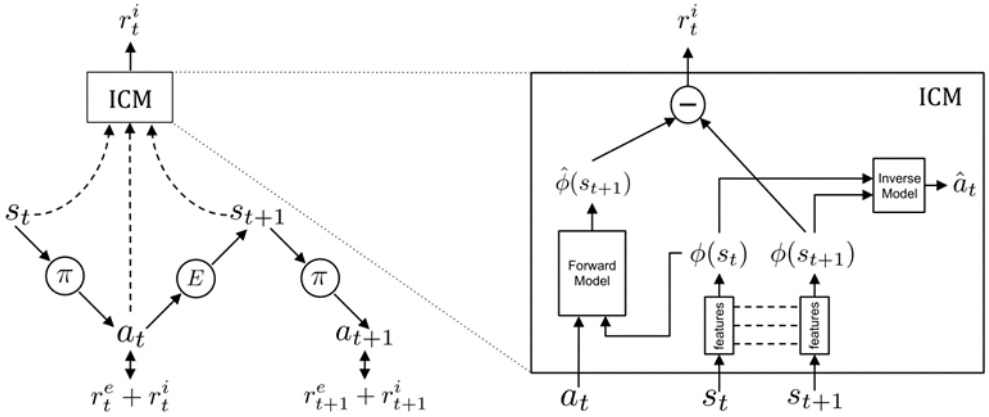


图 5.1 基于内在好奇心模型的深度强化学习算法结构图

5.2.3 策略优化

基于内在好奇心模型的深度强化学习方法的优化是一个整体的优化问题，由公式（5-1）、公式（5-3）以及公式（5-5）组成，表示为：

$$min_{\theta_P,\theta_I,\theta_F}[-\lambda\mathbb{E}\left[\nabla_\theta log\,\pi_{\theta_P}(a_t|s_t)\,\hat{A}_t\right]+(1-\beta)L_I+\beta L_F] \tag{5-7}$$

其中$0 \le \beta \le 1$是一个超参数，它权衡反向模型的损失相对内在好奇心模型整体损失的重要性。$\lambda > 0$是一个超参数，权衡策略梯度损失对策略学习的重要性。

5.3 基于好奇心的深度强化学习算法的实现

5.3.1 内在好奇心模型的实现

基于好奇心奖励地深度强化学习结构如图 5.1 所示，从图中可以看出，内在好奇心模块由一个可以当前时刻动作与状态输出下一时刻预测状态的正向模型(Forward Model)，以及一个对状态进行特征提取的反向模型(Inverse Model)。本文使用一个神经网络来实现正向模型，两个神经网络来实现反向模型。

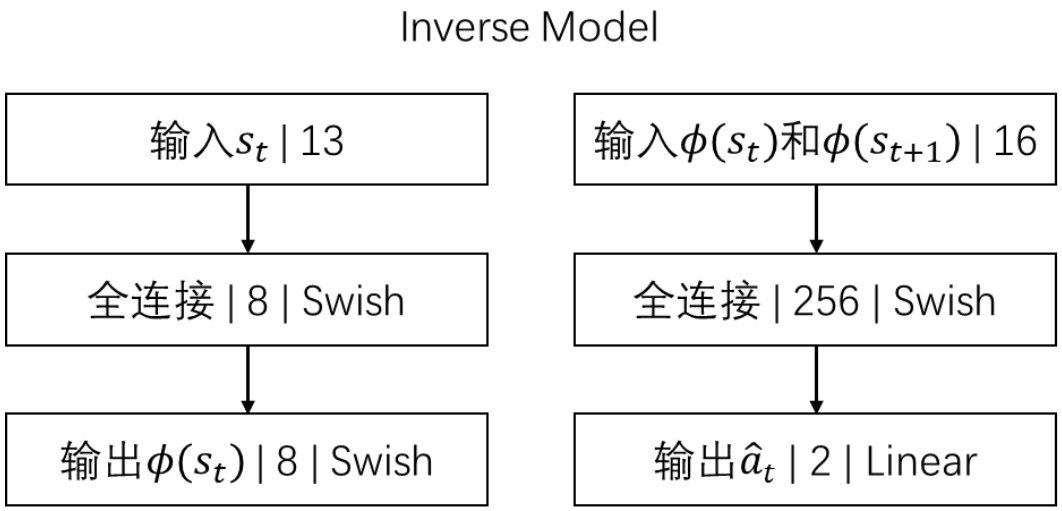


图 5.2 反向模型结构图

反向模型的神经网络结构如图 5.2 所示，该模型包括两个神经网络。一个神经网络输入 13 个维度的当前状态s_t，输出一个 8 个维度的$\phi(s_t)$，这是一个具有一个隐藏层的全连接神经网络，隐藏层具有 8 个隐藏节点，每一层之间均使用 Swish 激活函数，为了防止该神经网络改变输入的尺度，本文使用 TensorFlow 中的 variance scaling initializer 函数进行参数初始化。第一个网络输出的$\phi(s_t)$与$\phi(s_{t+1})$组合成一个 16 个维度的特征状态空间，输入第二个神经网络，输出两个维度的动作估计$\hat{a}_t$。第二个神经网络是一个单

隐层的全连接神经网络，输入层与隐层之间使用 Swish 激活函数，隐层与输出层之间使用线性激活函数，隐层含有 256 个隐藏节点。神经网络的计算过程可以表示为公式(4-14)。

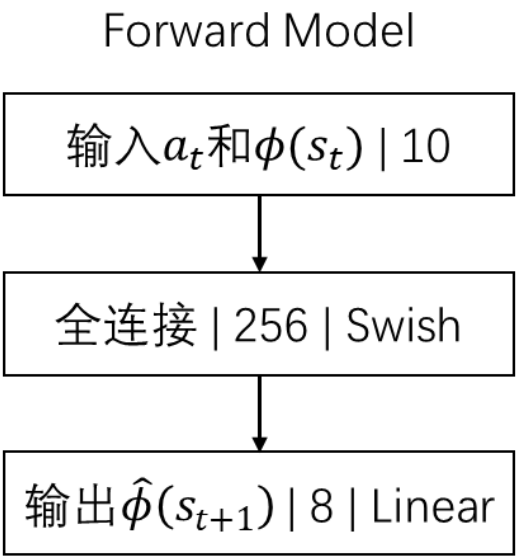


图 5.3 正向模型结构图

正向模型的神经网络结构如图 5.3 所示，该模型包括一个神经网络。神经网络输入当前时刻的动作a_t以及当前时刻的特征状态空间$\phi(s_t)$，共 10 个维度，输出对下一时刻的特征状态空间的估计$\hat{\phi}(s_{t+1})$，共 8 个维度。该神经网络具有一个隐层，隐层具有 256 个隐藏节点。输入层与隐层之间使用 Swish 激活函数，隐层与输出层之间使用线性激活函数。神经网络的计算过程可以表示为公式（4-14）。

5.3.2 深度强化学习算法实现

深度强化学习算法实现主要包括算法的实现，策略梯度的优化方法，以及奖励函数的设计等。在基于好奇心奖励的深度强化学习中，同样使用了基于 Actor-Critic 框架的算法实现，在 Actor-Critic 框架的价值函数选择上以及优化方法上，使用了近端优化策略。Actor 与 Critic 网络结构与前文使用的结构相同，如图 4.3 所示。奖励函数的设计方面，基于好奇心奖励的深度强化学习中的奖励函数与前文中使用的奖励函数一致，可以表示为公式（4-15）。

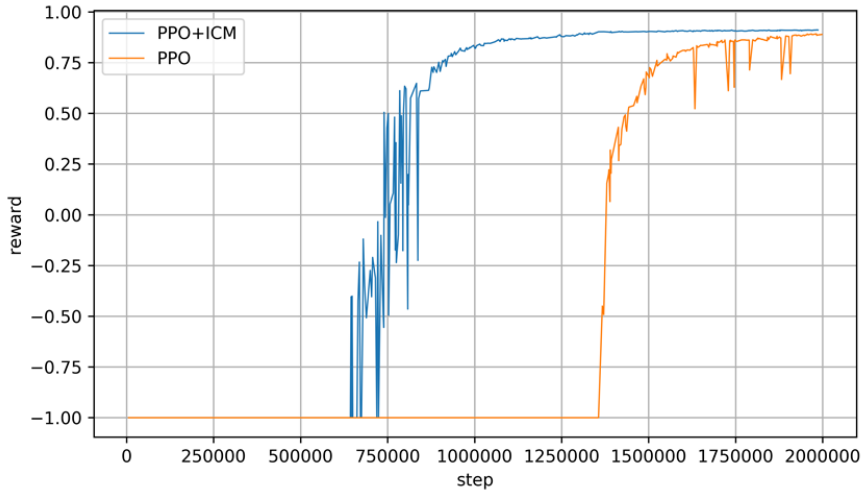
5.3.3 算法实现

综上所述，本文提出的基于好奇心奖励的深度强化学习的水下机器人未知环境运动规划算法如算法 2 所示。

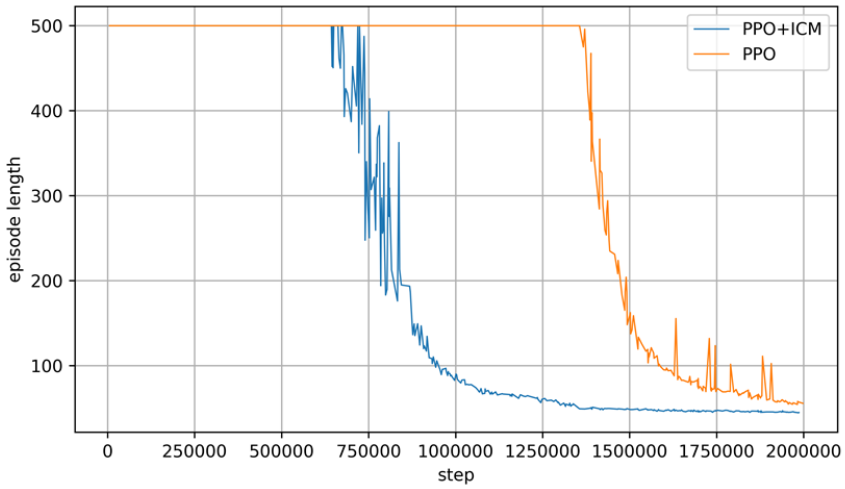
算法 2 基于好奇心奖励的水下机器人未知环境运动规划算法

1：初始化 Actor 网络与 Critic 网络参数θ以及ϕ；
2：初始化特征提取网络、前向模型以及反向模型的参数；
3：初始化训练环境；
4：AUV 根据策略$\pi_\theta(a_t|s_t)$以及状态s_t选择动作a_t；
5：观察执行a_t后 AUV 的状态s_{t+1}，并根据s_{t+1}与奖励函数得到奖励r_t^e；
6：通过好奇心模型获得好奇心奖励r_t^i；
7：计算总奖励$r_t = r_t^i + r_t^e$；
8：收集数据集(s_t, a_t, r_t)并将其保存在内存D中；
9：判断s_{t+1}是否为终止状态，如果是，进行 11，不是则返回 4；
10：判断当前训练回合是否达到规定的最大步数t_{max}，如果是，进行 11，不是则返回 4；
11：使用公式（4-11）计算优势函数；
12：通过公式（4-10），使用梯度法优化 Actor 策略网络参数θ；
13：通过公式（4-12），使用梯度法优化 Critic 评价网络参数ϕ；
14：通过公式（5-3）以及公式（5-5）计算好奇心模型损失；
15：通过公式（5-7），使用梯度法优化好奇心模型内的神经网络参数；
16：更新策略$\pi_{old} \leftarrow \pi_\theta$；
17：判断当前训练总步数是否达到训练要求的步数，如果是，进行 18，不是则返回 3；
18：算法结束。

为了说明基于好奇心奖励的深度强化学习算法的有效性，本文将其与无好奇心奖励的算法训练的模型进行了对比。本文使用了一个单独的训练环境，让 AUV 寻找环境中的目标点，目标点随机出现但距离 AUV 的初始距离均大于 200m 且小于 300m。图中展示了两种方法的训练结果。



(a) 奖励



(b) 回合步数

图 5.4 基于好奇心奖励与无好奇心奖励训练过程对比图

图 5.4 (a)是两种算法可以获得的奖励随着训练进行变化的折线图，图 5.4 (b)是两种算法每回合所需要的步数随着训练进行变化的折线图。从两图中均可以看出，基于好奇心奖励的训练速度更快。在智能体所需要探索的环境状态空间非常大时，没有中间奖励的深度强化学习算法的训练明显变得更加困难，它难以抵达目标点；而基于好奇心奖励的模型，由于对其探索过程具有奖励，因此可以更快的探索未知的环境状态，从而更快的实现更大环境的目标寻找，最终完成训练。

5.4 基于课程学习的训练

本文使用 TensorFlow 进行的神经网络搭建，使用 Adam 优化器进行优化，使用 Titan V 显示处理器搭配 i9 7980XE 处理器，训练时间 20 小时，训练的具体参数如表 5.1 所示。课程设计与没有好奇心奖励算法课程一致，具体描述可以参考 4.3.2 节。

表 5.1 训练参数

参数	值
学习率	0.0003
γ	0.99
η	0.01
β	0.2
λ	0.1
课程 1 训练步数	100000
课程 2 训练步数	900000
r_a	1
r_b	-1

t_{max}	200

基于好奇心奖励的模型在经过课程 1 与课程 2 的完整训练后，在课程 2 环境中规划的轨迹如图 5.5 所示，机器人位于环境重心，初始朝向右侧。为了更好的对比，选取目标点时选择了与第四章模型中相同的目标点位置。从图中可以看出，轨迹与图 4.5 有一定的区别，在这里 AUV 会首先加速再转向，然后朝向目标点直线前进，发现障碍物后左转进行回避，最终抵达目标点。对该轨迹进行定量分析，规划所需要的步数为 94 步；规划所需要的时间为 47.18s；规划的轨迹总距离为 107.86m。图 5.6 为定量分析的对比图，可以看出，使用好奇心规划的模型在规划时间上存在一定优势。但是规划的轨迹较长。本文在训练时所使用的奖励函数中，对规划时间进行了惩罚，而好奇心模型的规划时间更短，这意味着基于好奇心奖励的模型在本文的条件下有更好的表现。

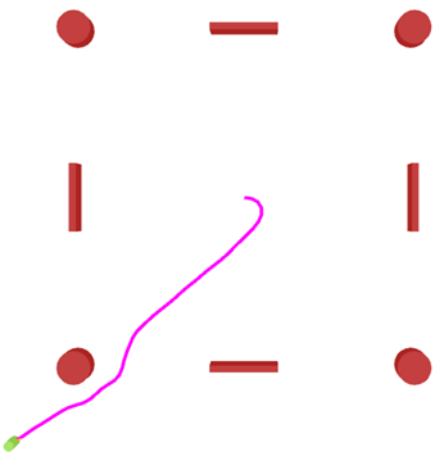


图 5.5 基于好奇心奖励深度强化学习模型课程 2 规划轨迹示意图

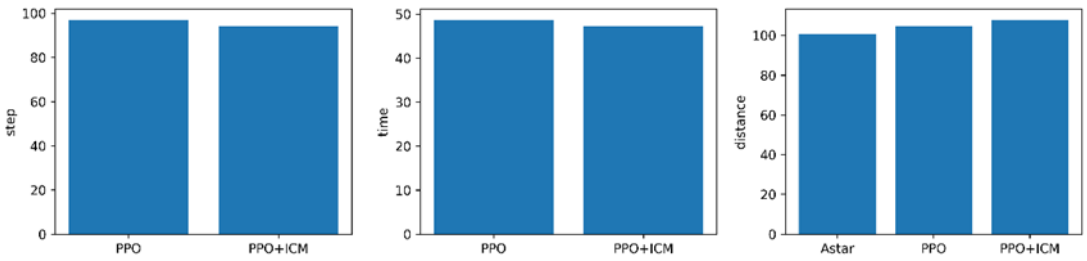
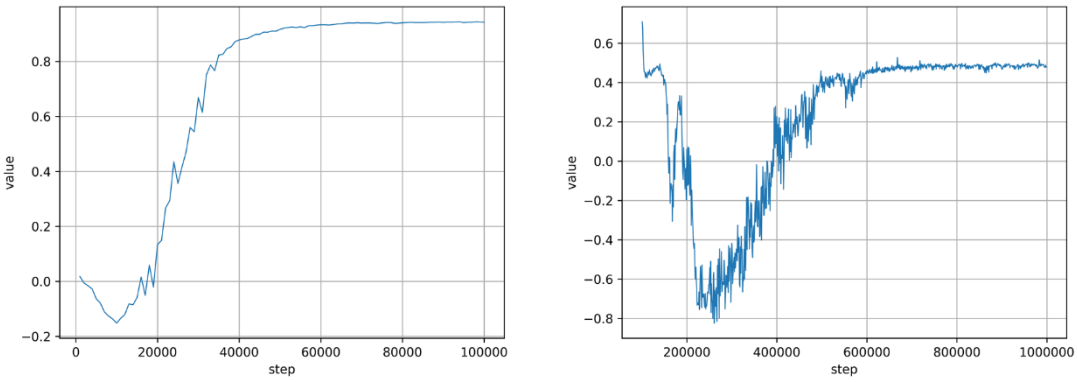


图 5.6 算法在课程 2 环境中步数、时间以及距离柱形图

图 5.7 (a)为模型在课程 1 中的价值，图 5.8 (a)中为模型在课程 1 时的回合长度，从两张图中可以看出，在课程 1 训练开始时，模型价值为负，回合长度接近t_{max}，这说明 AUV 无法抵达目标点，随着训练的进行，价值迅速增加，回合长度迅速降低，直到步数达到 40000 次时，价值与回合长度均趋近于收敛。这说明在没有障碍物的情况下，模

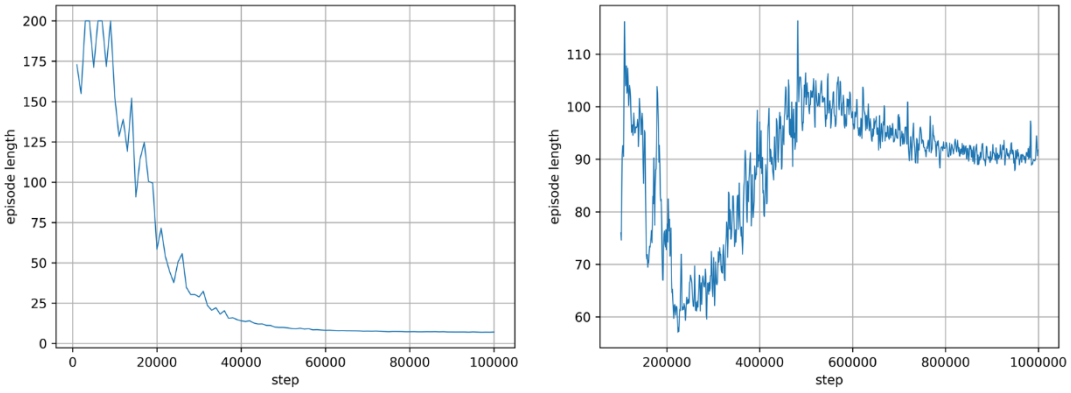
型可以很快找到抵达目标点的策略。

图 5.7 (b)为模型在课程 2 中的价值，图 5.8 (b)中为模型在课程 1 中的回合长度，从两张图中可以看出，课程 2 训练开始时，AUV 会与障碍物发生碰撞，价值降低，回合长度降低，随着训练的进行，价值增加，回合长度增加，这说明 AUV 找到了躲避障碍物的策略。训练次数在 500000-600000 次之间时，价值增加，回合长度降低，说明模型开始寻找更加节省时间的策略在训练次数到达 600000 次时，价值趋于收敛，模型找到了完成运动规划任务所需的策略。



(a) 课程1训练价值　　(b) 课程2训练价值

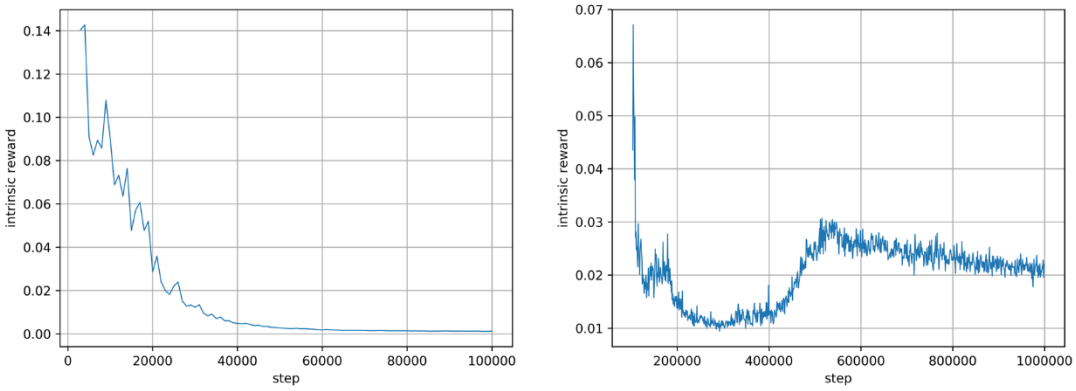
图 5.7 基于好奇心奖励的深度强化学习模型价值折线图



(a) 课程1回合步数　　(b) 课程2回合步数

图 5.8 基于好奇心奖励的深度强化学习回合步数折线图

图 5.9 为模型在训练过程中的好奇心奖励以及前向模型与后向模型的损失。可以看到，在课程 1 的训练过程中，整个好奇心模型均趋于收敛，好奇心奖励快速降低。在课程 2 的训练过程中，在 600000 步训练之后好奇心模型趋于收敛，这与课程 2 的训练过程相对应，在找到最优策略前，好奇心模型并不稳定，较高的好奇心奖励可以帮助模型探索未知的状态空间。在策略收敛后，好奇心作用逐渐降低，好奇心模型趋于收敛。



(a) 课程1内在奖励　　(b) 课程2内在奖励

图 5.9 内在奖励折线图

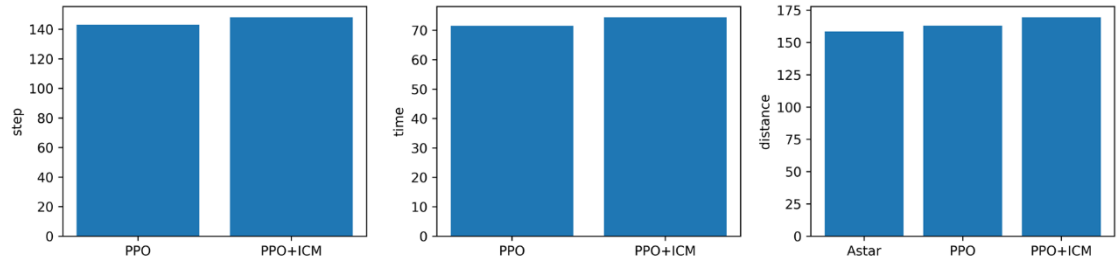
5.5 未知环境仿真试验及结果分析

本文将提出的方法迁移至其他的未知环境中进行试验，包括一个单目标点的未知环境以及一个多目标点的未知环境。环境模型如图 4.9 所示，其详细描述可以参考 4.4.1 节。规划的轨迹如图 5.10 所示。

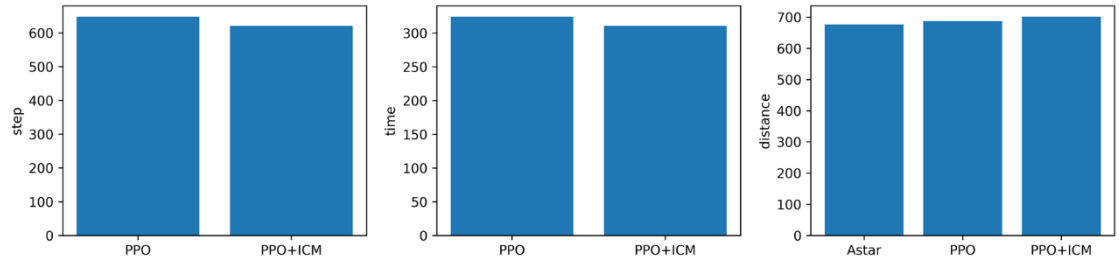
(a) 单目标点未知环境　　(b) 多目标点未知环境

图 5.10 基于好奇心奖励的深度强化学习模型于未知环境规划轨迹图

从图中可以看出，基于好奇心奖励的模型可以顺利的躲避障碍物并且抵达目标点。在多目标未知环境中，AUV 规划的轨迹与无好奇心模型规划的轨迹有很大区别，这里的轨迹在抵达障碍物后并没有选择减速，而是在保持速度的情况下进行转向，这与在课程 2 训练环境中轨迹特征相似。好奇心模型在单目标点环境中规划的轨迹距离为 169.36m，规划所需步数为 148 步，规划时间为 74.34s；在多目标点环境中规划的轨迹距离为 701.49m，规划所需步数为 621 步，规划时间为 310.94s。与无好奇心模型和 A* 算法的对比如图 5.11 所示。



(a) 单目标未知环境



(b) 多目标未知环境

图 5.11 算法在未知环境中规划步数、时间以及距离柱形图

图中可以看出，在单目标未知环境中，无好奇心模型表现非常优秀，三个评价指标均优于好奇心模型，而且规划的轨迹距离仅比 A*算法规划的距离多 2.9%，轨迹非常接近与全局最优路径。但是在多目标环境中，基于好奇心的模型表现更加优秀，在规划步数与规划时间上均优于无好奇心模型。这是由于好奇心模型中的 AUV 倾向于在保持速度的同时进行转向，无好奇心模型倾向于减速后转向导致的。

无好奇心的模型在学习过程中虽然会随机前往未知的状态，但是在使用神经网络对策略进行表示时，仍有可能陷入局部最优解。而基于好奇心的模型会对无法预测的状态保持好奇（给与好奇心奖励），这其中包括当前时刻非最优但实际更为优秀的状态，这样有助于模型尝试更多的状态，跳出局部最优，寻找更优秀的策略。由于好奇心模型会探索更多的状态，在连续状态空间中，状态维度过大，因此规划的轨迹还并不平滑，在单目标未知环境中规划的轨迹较差。对好奇心模型进行更长时间的训练可能可以解决这个问题，这是接下来的研究方向。

可以发现，无论是好奇心模型还是无好奇心模型，在对比 A*算法时，规划的轨迹都不够平滑，规划轨迹距离较长。本文的目标并不是寻找可以替代全局路径规划的方法，在大规模的环境中进行运动规划时，基于地图的规划仍然是更好的选择，无地图的规划应该为全局规划提供辅助，由于海洋环境的复杂多变，局部的无地图规划也更加重要。

5.6 本章小结

本章首先介绍了好奇心的背景与研究现状，然后详细介绍了好奇心奖励的推导与内在好奇模块(Intrinsic Curiosity Module，ICM)的结构。设计了基于好奇心奖励的深度强化学习算法，阐述了本文构建的算法的详细神经网络结构以及算法的完整流程。使用了课程学习的训练方法对算法进行了训练，训练的模型可以顺利完成训练环境的任务。最后在多目标点未知环境中对训练的模型进行了仿真试验，试验验证了该算法可以顺利的规划出未知环境的轨迹。

结论

运动规划技术是智能水下机器人的重要研究内容之一。由于海洋环境的未知与不确定性，以及 AUV 自身的执行机构能力限制，这个问题是一个多约束问题。为了解决未知海洋环境下的 AUV 运动规划问题，本文在深度学习与强化学习的理论基础上，实现了一种基于策略的 AUV 未知环境运动规划系统。研究内容主要包括以下几个方面：

（1）分析了水下机器人动力学模型，在水下机器人动力学模型的基础上，对多约束的运动规划问题进行建模。AUV 运动规划器将传感器信息作为输入，输出目标速度，再结合 S 面控制器进行推进器的控制，最终输出一系列的从起始位置抵达目标点位置的轨迹，完成未知环境下的运动规划任务；

（2）提出了一种基于深度强化学习的水下机器人未知环境的运动规划方法。针对基于值函数的强化学习难以解决连续动作空间的问题，以及在解决多自由度规划时的“维度灾难”问题，本文进行了改进，使用了基于策略的近端策略优化方法来实现深度强化学习模型。它直接输出运动规划策略，算法的效率更高，而且解决了基于值函数的强化学习无法解决重名状态的问题。在训练深度强化学习的过程中，对训练方法进行了改进，基于课程学习的思想，提出了一种适用于水下机器人的课程构建方法。使用人工知识的方法对课程进行构建，课程分别训练 AUV 寻找目标点以及躲避障碍物的能力，并且逐渐提高训练的难度。仿真试验证明，课程训练方法提高了大幅提高了训练的速度，解决了连续动作空间中奖励稀疏的问题。除此之外，本文在未知环境中对算法模型进行了仿真试验，仿真试验结果证明了模型具有迁移至未知环境的能力；

（3）在基于课程训练的基础上，本文进一步提出了一种基于好奇心奖励的深度强化学习方法。本文的算法中使用深度神经网络来输出策略，这简化了算法的空间复杂度，但是算法有可能陷入局部最优解。为了解决这个问题，本文在奖励函数中加入了好奇心奖励，好奇心奖励是一种对环境状态空间预测的偏差，意味着奖励鼓励 AUV 尝试无法预测的状态，以跳出局部最优。仿真试验证明了基于好奇心奖励训练的算法模型的有效性。除此之外，本文在未知环境中对算法模型进行了仿真试验，仿真试验结果证明了该模型同样具有迁移至未知环境的能力。

尽管本文在 AUV 未知环境下的运动规划方面有着良好的进展，但已经完成的研究工作还有待于进一步深入，作者对今后研究工作提出以下几点展望：

（1）本研究对提出的 AUV 的运动规划系统进行了充足的仿真试验，但是由于硬件

条件限制，该系统还没有进行实际实验，在实际中本文所构建的规划系统仍有待进一步的验证；

（2）对基于好奇心奖励的模型，无好奇心奖励的模型以及使用 A*算法规划的轨迹进行了对比讨论，发现基于好奇心奖励的模型可以获得更多的时间奖励，规划出的轨迹在时间上更加优秀。与 A*算法的全局最优轨迹相比，由于 A*算法已知环境地图，深度强化学习在轨迹距离上与 A*对比仍存在一定的差距。将 A*算法与深度强化学习进行结合是未来的研究方向；

（3）海洋环境中的海流对运动规划存在一定的影响，本文中没有对海流进行考虑，有待进一步添加补充。此外 AUV 是一种 6 自由度的机器人，完整考虑其 6 自由度进行运动规划也是进一步研究的方向。

参考文献

[1] 徐玉如，庞永杰，甘永,等. 智能水下机器人技术展望[J]. 智能系统学报，2006, 1(01):16-23.

[2] 李晔. 微小型水下机器人运动控制技术研究[D]. 哈尔滨工程大学, 2007.

[3] Von Alt C. REMUS 100 transportable mine countermeasure package[C]// Oceans. IEEE, 2003:1925-1930 Vol.4.

[4] Prestero T. Development of a six-degree of freedom simulation model for the REMUS autonomous underwater vehicle[C]// Oceans. IEEE, 2002:450-455 vol.1.

[5] Anderson B, Crowell J. Workhorse AUV - A cost-sensible new Autonomous Underwater Vehicle for Surveys/Soundings, Search & Rescue, and Research[C]// Oceans. IEEE, 2005:1-6.

[6] Kondo H, Yu S, Ura T. Object observation in detail by the AUV "Tri-Dog 1" with laser pointers[C]// Oceans. IEEE, 2001:390-396 vol.1.

[7] 苏玉民，万磊，李晔,等. 舵桨联合操纵微小型水下机器人的开发[J]. 机器人，2007, 29(2):151-154.

[8] 陈超，唐坚. 基于可视图法的水面无人艇路径规划设计[J]. 中国造船，2013 (1): 129-135.

[9] Xiao H, Cui R, Xu D. A sampling-based bayesian approach for cooperative multiagent online search with resource constraints[J]. IEEE transactions on cybernetics, 2018, 48(6): 1773-1785.

[10] Li Y, Cui R, Li Z, et al. Neural Network Approximation-based Near-optimal Motion Planning with Kinodynamic Constraints Using RRT[J]. IEEE Transactions on Industrial Electronics, 2018.

[11] 庄佳园，张磊，孙寒冰，等. 应用改进随机树算法的无人艇局部路径规划[J]. 哈尔滨工业大学学报, 2015, 47(1): 112-117.

[12] 王芳，万磊，徐玉如，等. 基于改进人工势场的水下机器人路径规划[J]. 华中科技大学学报：自然科学版, 2011 (S2): 184-187.

[13] 李欣，朱大奇. 基于人工势场法的自治水下机器人路径规划[J]. 上海海事大学学报, 2010, 31(2): 35-39.

[14]俞建成, 李强, 张艾群, 等. 水下机器人的神经网络自适应控制[J]. 控制理论与应用, 2008, 25(1): 9-13.

[15]彭良, 卢迎春, 万磊, 等. 水下智能潜器的神经网络运动控制[J]. 海洋工程, 1995 (2): 38-46.

[16]Garau B, Alvarez A, Oliver G. Path Planning of Autonomous Underwater Vehicles in Current Fields with Complex Spatial Variability: an A* Approach[C]// IEEE International Conference on Robotics and Automation. IEEE, 2005:194-198.

[17]Alvarez A, Caiti A, Onken R. Evolutionary path planning for autonomous underwater vehicles in a variable ocean[J]. IEEE Journal of Oceanic Engineering, 2004, 29(2):418-429.

[18]Warren C W. A technique for autonomous underwater vehicle route planning[J]. IEEE Journal of Oceanic Engineering, 1990, 15(3):199-204.

[19]Sugihara K, Yuh J. GA-based motion planning for underwater robotic vehicles[C]// International Symposium on Unmanned Untethered Submersible Technology. UNIVERSITY OF NEW HAMPSHIRE-MARINE SYSTEMS, 1997: 406-415.

[20]Stentz A. Optimal and Efficient Path Planning for Unknown and Dynamic Environments[J]. International Journal of Robotics & Automation, 1993, 10(3):89--100.

[21]徐玉如, 姚耀中. 考虑海流影响的水下机器人全局路径规划研究[J]. 中国造船, 2008, 49(4):109-114.

[22]毛宇峰, 庞永杰. 改进粒子群在水下机器人路径规划中的应用[J]. 计算机应用, 2010, 30(3):789-792.

[23]刘利强, 于飞, 戴运桃. 基于蚁群算法的水下潜器三维空间路径规划[J]. 系统仿真学报, 2008, 20(14):3712-3716.

[24]刘利强, 戴运桃, 王丽华, 等. 基于蚁群算法的水下潜器全局路径规划技术研究[J]. 系统仿真学报, 2007, 19(18): 4174-4177.

[25]张汝波, 周宁, 顾国昌,等. 基于强化学习的智能机器人避碰方法研究[J]. 机器人, 1999, 21(3):204-209.

[26]杨广铭, 张汝波, 顾国昌. 基于 Q-learning 的机器人避碰控制方法的研究[J]. 哈尔滨工程大学学报, 1999, 20(5):77-82.

[27]Li J H, Lee M J, Park S H, et al. Real time path planning for a class of torpedo-type

AUVs in unknown environment[C]// Autonomous Underwater Vehicles. IEEE, 2012:1-6.

[28] Moore A W, Atkeson C G. The parti-game algorithm for variable resolution reinforcement learning in multidimensional state-spaces[J]. Machine Learning, 1995, 21(3):199-233.

[29] Vamvoudakis K G, Vrabie D, Lewis F L. Online adaptive algorithm for optimal control with integral reinforcement learning[J]. International Journal of Robust & Nonlinear Control, 2015, 24(17):2686-2710.

[30] Mnih V, Kavukcuoglu K, Silver D, et al. Playing atari with deep reinforcement learning[J]. arXiv preprint arXiv:1312.5602, 2013.

[31] Mnih V, Kavukcuoglu K, Silver D, et al. Human-level control through deep reinforcement learning[J]. Nature, 2015, 518(7540): 529.

[32] Cui R, Yang C, Li Y, et al. Adaptive Neural Network Control of AUVs With Control Input Nonlinearities Using Reinforcement Learning[J]. IEEE Transactions on Systems Man & Cybernetics Systems, 2017, 47(6):1019-1029.

[33] BELLMAN R. Dynamic programming and Lagrange multipliers [J]. Proceedings of the National Academy of Sciences, 1956, 42(10): 767 – 769.

[34] WERBOS P J. Advanced forecasting methods for global crisis warning and models of intelligence [J]. General Systems Yearbook, 1977, 22(12): 25 – 38.

[35] SUTTON R S, BARTO A G. Reinforcement Learning: An Introduction [M]. Cambridge MA: MIT Press, 1998.

[36] WATKINS C J C H. Learning from delayed rewards [D]. Cambridge: University of Cambridge, 1989.

[37] RUMMERY G A, NIRANJAN M. On-Line Q-Learning Using Connectionist Systems [M]. Cambridge: University of Cambridge, Department of Engineering, 1994.

[38] BERTSEKAS D P, TSITSIKLIS J N. Neuro-dynamic programming: an overview [C] //Proceedings of the 34th IEEE Conference on Decision and Control. New Orleans: IEEE, 1995, 1: 560 – 564.

[39] THRUN S. Monte Carlo POMDPs [C] // Advances in Neural Information Processing Systems. Denver: MIT Press, 1999, 12: 1064–1070.

[40] KOCSIS L, SZEPESVARI C. Bandit based Monte-Carlo planning [C] //Proceedings of the European Conference on Machine Learning. Berlin: Springer, 2006: 282–293.

[41]LEWIS F L, VRABIE D. Reinforcement learning and adaptive dynamic programming for feedback control[J].IEEE Circuits and Systems Magazine, 2009, 9(3): 32 – 50.

[42]SILVER D, LEVER G, HEESS N, et al. Deterministic policy gradient algorithms [C] //Proceedings of the International Conference on Machine Learning. Beijing: ACM, 2014: 387 – 395.

[43]Kawano H, Ura T. Motion planning algorithm for nonholonomic autonomous underwater vehicle in disturbance using reinforcement learning and teaching method[C]// IEEE International Conference on Robotics & Automation. IEEE, 2002.

[44]Carreras M, Batlle J, Ridao P. Hybrid coordination of reinforcement learning-based behaviors for AUV control[C]// IEEE/RSJ International Conference on Intelligent Robots & Systems. IEEE, 2001.

[45]Carreras, M. Yuh, J. Batlle, J. Pere Ridao. A Behavior-Based Scheme Using Reinforcement Learning for Autonomous Underwater Vehicles[J]. IEEE Journal of Oceanic Engineering, 2005, 30(2):416-427.

[46]Sencianes E F, Andrés, Carreras Pérez, et al. Policy gradient based Reinforcement Learning for real autonomous underwater cable tracking[C]// IEEE/RSJ International Conference on Intelligent Robots & Systems. IEEE, 2008.

[47]El-Fakdi A, Carreras M. Two-step gradient-based reinforcement learning for underwater robotics behavior learning[J]. Robotics & Autonomous Systems, 2013, 61(3):271-282.

[48]Tai L, Liu M. A robot exploration strategy based on Q-learning network[C]// 2016 IEEE International Conference on Real-time Computing and Robotics (RCAR). IEEE, 2016.

[49]Xie C, Patil S, Moldovan T, et al. Model-based Reinforcement Learning with Parametrized Physical Models and Optimism-Driven Exploration[C]// IEEE International Conference on Robotics & Automation. IEEE, 2016.

[50]Ng A Y, Coates A, Diel M, et al. Autonomous inverted helicopter flight via reinforcement learning[M]//Experimental Robotics IX. Springer, Berlin, Heidelberg, 2006: 363-372.

[51]HINTON G E, OSINDERO S, TEH Y W. A fast learning algorithm for deep belief nets [J]. Neural Computation, 2006, 18(7): 1527 –1554.

[52]ABDEL-HAMID O, MOHAMED A, JIANG H, et al. Convolutional neural networks for speech recognition [J]. IEEE/ACM Transactions on Audio, Speech, and Language

Processing, 2014, 22(10): 1533 –1545.

[53]CARLSON B A, CLEMENTS M A. A projection-based likelihood measure for speech recognition in noise [J]. IEEE Transactions on Speech and Audio Processing, 1994, 2(1): 97 – 102.

[54]OUYANG W, ZENG X, WANG X. Learning mutual visibility relationship for pedestrian detection with a deep model [J]. International Journal of Computer Vision, 2016, DOI: 10.1007/s11263-016-0890-9.

[55]KRIZHEVSKY A, SUTSKEVER I, HINTON G E. Imagenet classification with deep convolutional neural networks [C] //Advances in Neural Information Processing Systems. Lake Tahoe: MIT Press, 2012: 1097 – 1105.

[56]GRAVERS A, MOHAMED A, HINTON G. Speech recognition with deep recurrent neural networks [C] //Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing. Vancouver: IEEE, 2013: 6645 – 6649.

[57]XU K, BA J, KIROS R, et al. Show, attend and tell: neural image caption generation with visual attention [C] //Proceedings of the 32nd International Conference on Machine Learning. Lille: ACM, 2015: 2048 – 2057.

[58]PINHEIRO P, COLLOBERT R. Recurrent convolutional neural networks for scene labeling [C] //Proceedings of the 31nd International Conference on Machine Learning. Beijing: ACM, 2014: 82 – 90.

[59]HE K M, ZHANG X, REN S, et al. Deep residual learning for image recognition [C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016.

[60]Lecun Y, Muller U, Ben J, et al. Off-road obstacle avoidance through end-to-end learning[C]// International Conference on Neural Information Processing Systems. MIT Press, 2005:739-746.

[61]Chen C, Seff A, Kornhauser A, et al. DeepDriving: Learning Affordance for Direct Perception in Autonomous Driving[C]// IEEE International Conference on Computer Vision. IEEE Computer Society, 2015:2722-2730.

[62]Pfeiffer M, Schaeuble M, Nieto J, et al. From perception to decision: A data-driven approach to end-to-end motion planning for autonomous ground robots[C]// IEEE

International Conference on Robotics and Automation. IEEE, 2017:1527-1533.

[63] Li J H, Lee P M. A neural network adaptive controller design for free-pitch-angle diving behavior of an autonomous underwater vehicle[J]. Robotics & Autonomous Systems, 2005, 52(2):132-147.

[64] Zhang L J, Qi X, Pang Y J. Adaptive output feedback control based on DRFNN for AUV[J]. Ocean Engineering, 2009, 36(9):716-722.

[65] McCulloch, Warren S, Pitts, et al. A logical calculus of the ideas immanent in nervous activity[J]. Bulletin of Mathematical Biology, 1943, 5(4):115-133.

[66] SHIBATA K, IIDA M. Acquisition of box pushing by direct-vision-based reinforcement learning [C] //Proceedings of the SICE Annual Conference. Nagoya: IEEE, 2003, 3: 2322 – 2327.

[67] SHIBATA K, OKABE Y. Reinforcement learning when visual sensory signals are directly given as inputs [C] //Proceedings of the International Conference on Neural Networks. Houston: IEEE, 1997, 3: 1716 – 1720.

[68] LANGE S, RIEDMILLER M, VOIGTLANDER A. Autonomous reinforcement learning on raw visual input data in a real world application [C] //Proceedings of the International Joint Conference on Neural Networks. Brisbane: IEEE, 2012: 1 – 8.

[69] KOUTNIK J, SCHMIDHUBER J, GOMEZ F. Online evolution of deep convolutional network for vision-based reinforcement learning [M] //From Animals to Animats 13. New York: Springer, 2014: 260 – 269.

[70] Zhang F, Leitner J, Milford M, et al. Towards vision-based deep reinforcement learning for robotic motion control[J]. arXiv preprint arXiv:1511.03791, 2015.

[71] Gu S, Holly E, Lillicrap T, et al. Deep Reinforcement Learning for Robotic Manipulation with Asynchronous Off-Policy Updates[C]// IEEE International Conference on Robotics & Automation. IEEE, 2017.

[72] Tai L, Paolo G, Liu M. Virtual-to-real deep reinforcement learning: Continuous control of mobile robots for mapless navigation[C]// Ieee/rsj International Conference on Intelligent Robots and Systems. IEEE, 2017.

[73] Fossen T I. Handbook of Marine Craft Hydrodynamics and Motion Control[J]. IEEE Control Systems, 2016, 36(1):78-79.

[74]Lillicrap T P, Hunt J J, Pritzel A, et al. Continuous control with deep reinforcement learning[J]. arXiv preprint arXiv:1509.02971, 2015.

[75]Mnih V, Badia A P, Mirza M, et al. Asynchronous methods for deep reinforcement learning[C]//International conference on machine learning. 2016: 1928-1937.

[76]Wu Y, Mansimov E, Grosse R B, et al. Scalable trust-region method for deep reinforcement learning using kronecker-factored approximation[C]//Advances in neural information processing systems. 2017: 5279-5288.

[77]Schulman J, Wolski F, Dhariwal P, et al. Proximal policy optimization algorithms[J]. arXiv preprint arXiv:1707.06347, 2017.

[78]Schulman J, Levine S, Abbeel P, et al. Trust region policy optimization[C]//International Conference on Machine Learning. 2015: 1889-1897.

[79]Kakade S M, Langford J. Approximately Optimal Approximate Reinforcement Learning[C]// Nineteenth International Conference on Machine Learning. Morgan Kaufmann Publishers Inc. 2002.

[80]Ramachandran P, Zoph B, Le Q V. Swish: a self-gated activation function[J]. arXiv preprint arXiv:1710.05941, 2017.

[81]Bengio Y, Jérôme Louradour, Collobert R, et al. Curriculum learning[C]// International Conference on Machine Learning. ACM, 2009.

[82]Khan F, Mutlu B, Zhu X. How do humans teach: On curriculum learning and teaching dimension[C]//Advances in Neural Information Processing Systems. 2011: 1449-1457.

[83]Basu S, Christensen J. Teaching Classification Boundaries to Humans[C]//AAAI. 2013.

[84]Bengio Y, Courville A, Vincent P. Representation learning: A review and new perspectives[J]. IEEE transactions on pattern analysis and machine intelligence, 2013, 35(8): 1798-1828.

[85]Bengio Y. Evolving culture versus local minima[M]//Growing Adaptive Machines. Springer, Berlin, Heidelberg, 2014: 109-138.

[86]Spitkovsky V I, Alshawi H, Jurafsky D. Baby Steps: How “Less is More” in unsupervised dependency parsing[J]. NIPS: Grammar Induction, Representation of Language and Language Learning, 2009: 1-10.

[87]Tian Y, Gong Q, Shang W, et al. Elf: An extensive, lightweight and flexible research

platform for real-time strategy games[C]//Advances in Neural Information Processing Systems. 2017: 2659-2669.

[88] Pathak D, Agrawal P, Efros A A, et al. Curiosity-driven exploration by self-supervised prediction[C]//International Conference on Machine Learning (ICML). 2017, 2017.

[89] Bellemare M, Srinivasan S, Ostrovski G, et al. Unifying count-based exploration and intrinsic motivation[C]//Advances in Neural Information Processing Systems. 2016: 1471-1479.

[90] Lopes M, Lang T, Toussaint M, et al. Exploration in model-based reinforcement learning by empirically estimating learning progress[C]// Advances in Neural Information Processing Systems. 2012: 206-214.

[91] Poupart P, Vlassis N, Hoey J, et al. An analytic solution to discrete Bayesian reinforcement learning[C]// International Conference on Machine Learning. ACM, 2006.

[92] Houthooft R, Chen X, Duan Y, et al. Vime: Variational information maximizing exploration[C]//Advances in Neural Information Processing Systems. 2016: 1109-1117.

[93] Mohamed S, Rezende D J. Variational information maximisation for intrinsically motivated reinforcement learning[C]//Advances in neural information processing systems. 2015: 2125-2133.

[94] Meyer J A, Wilson S W. A Possibility for Implementing Curiosity and Boredom in Model-Building Neural Controllers[C]// International Conference on Simulation of Adaptive Behavior on from Animals to Animats. MIT Press, 1991.

[95] Singh S, Lewis R L, Barto A G, et al. Intrinsically Motivated Reinforcement Learning: An Evolutionary Perspective[J]. IEEE Transactions on Autonomous Mental Development, 2010, 2(2):70-82.

[96] Stadie B C, Levine S, Abbeel P. Incentivizing exploration in reinforcement learning with deep predictive models[J]. arXiv preprint arXiv:1507.00814, 2015.

攻读硕士学位期间发表的论文和取得的科研成果

[1] 孙玉山，程俊涵，张国成等. 一种基于辅助决策系统的水下潜器路径规划方法 申请号：201810248836.1

[2] 张国成，程俊涵，孙玉山等. 一种基于多约束目标的水下机器人运动规划方法 申请号：201810764979.8

致谢

时间如白驹过隙，两年半的硕士研究生生活已经接近尾声，借此毕业论文完成之际，我想在这里感谢在这段时间以来给予过我帮助与关心的人。

首先，我要感谢我的导师孙玉山教授，孙老师的学术水平与人格魅力给我留下了很深刻的印象。在学习方面，孙老师严谨、认真、学识渊博，在毕业论文的撰写上对我提出了很多宝贵意见。在生活方面，孙老师态度随和、平易近人，百忙之中不忘关心学生，我也受到了孙老师的很多照顾。在工作方面，孙老师勤劳与务实的工作态度为我们树立了良好的榜样，这让我受益终生。

感谢苏玉民教授、庞永杰教授、秦洪德教授、万磊研究员、李晔教授等为我们创造的优秀的学习工作环境。感谢张国成老师对我研究生课题的指导，在两年半的学习生活中，张老师不仅通过言传身教，教会了我很多知识，同时也对我进行了很多批评与督促，帮助我不断地进步。

感谢冉祥瑞师兄、徐昊师兄、王相斌师兄以及陈庆龙师兄对我学习生活上的帮助，师兄们乐于助人，风趣幽默，帮助我学会了很多，在生活上也给予了我很多关怀。感谢吴凡宇同学、贾晨凯同学、王子楷同学、焦文龙同学、王立峰同学、封飞翔同学，他们与我一同度过了硕士两年半的时间，是同学更是朋友。感谢实验室所有的老师与同学们，很开心可以在这里与你们一起学习与工作。

感谢我的家人，在背后理解、支持和鼓励我，是你们无私的关心和帮助才成就了今天的我。

最后感谢学校、感谢实验室、感谢所有帮助和关心过我的人，谢谢你们！