doi: 10.7690/bgzdh.2024.01.018

一种基于强化学习的指挥智能体控制方法

林九根，朱衍明，余景锋，宋家平，吴如悦
（中国船舶工业系统工程研究院智能中心，北京 100094）

摘要：针对无人水下航行器（unmanned underwater vehicle，UUV）如何进行任务分配、航路规划、指挥控制问题，提出一种新的控制实现方法。搭建 UUV 指挥智能体训练平台，设计学习训练所需的想定，进行状态设计、数据适配、决策解析和规则库建立，选定近端策略优化（proximal policy optimization，PPO）强化学习算法进行训练，并进行应用验证。结果表明：指挥智能体能有效对 UUV 进行任务分配、航路规划、指挥控制；通过不断优化算法，可提高战胜基于规则的传统控制方法的胜率。

关键词：航路规划；任务分配；智能体；强化学习
中图分类号：TJ6; E925　　**文献标志码：**A

A Control Method of Command Agent Based on Reinforcement Learning

Lin Jiugen, Zhu Yanming, Yu Jingfeng, Song Jiaping, Wu Ruyue
(*Ai Department*, *CSSC Systems Engineering Research Institute*, *Beijing* 100094, *China*)

Abstract: Aiming at the methods of task allocation, route planning and command control of unmanned underwater vehicle (UUV), a new control implementation method, command agent based on deep reinforcement learning, is proposed to replace human in the loop or automatic command and control. Build UUV command agent training platform, design scenarios required for learning and training, conduct state design, data adaptation, decision analysis and rule base establishment, and select proximal policy optimization (PPO) reinforcement learning algorithm for training. The application verification of the command agent generated by training and learning is carried out. The results show that the command intelligence can effectively carry out task allocation, route planning, command and control of UUV, and make bold guesses. By continuously optimizing the algorithm, the winning rate of defeating the traditional rule-based control method can be improved.

Keywords: route planning; task allocation; agent; reinforcement learning

0　引言

作为一个复杂系统，无人水下航行器（UUV）近年来成为研究热点。无人水下航行器任务分配、航路规划、指挥控制是需要重点关注的关键问题[1-4]。UUA 的控制有其特殊性，需要具备以下能力：1) 与其他无人装备相比，由于水下通信受限，不具备“人在回路”条件，需要 UUV 在任务规划、路径规划、集群控制上具备更高的自主决策能力；2) 水下环境恶劣，需要面对洋流、断层及海底复杂地形，UUV 一旦失控也更难恢复或回收，因此需要具备更广泛的环境适应能力或抗干扰能力；3) 在编队控制中由于特殊的水下通信条件和工作环境，单个 UUV 的感知能力范围小且不全面，需要 UUV 集群具备协同指挥作业能力。笔者采用基于深度强化学习的训练方法，通过海量的不同场景进行学习训练，得到充分考虑环境变化下执行最优决策的指挥智能体。通过数据驱动的方式取代传统动力学推导方式，更优且快地解决 UUV 这类强耦合非线性模型的决策问题，提高其自主决策能力，提高 UUV 决策智能体的泛化能力，能够应对不同环境，提高其环境适应能力或抗干扰能力，从而解决弱通信约束条件下，UUV 的任务分配、航路规划和指挥控制问题，以及提高其协同指挥作业能力。

收稿日期：2023-09-20；**修回日期：**2023-10-25
第一作者：林九根（1983—），男，江西人，硕士。

1　构建指挥智能体训练平台

构建指挥智能体训练平台分为想定筹划和设计、指挥智能体设计、强化学习特征工程 3 部分。

1.1　想定筹划和设计

1.1.1　想定筹划

根据典型特定作战行动构想，将军事问题计算机化、模型化，并部署到仿真平台进行仿真推演运行。将作战构想中作战行动、作战规则、战场环境等信息数据化生成仿真数据，模拟战场兵力行动和战场环境布设，构建为指挥智能体训练平台的仿真系统部分。

1.1.2　想定设计

虚构一个作战场景，2020 年 8 月某军海上演习，

假定红方 3 条 UUV 进行反 Q(蓝方)。

1.1.2.1　红方作战目标

出动 3 条 UUV，对蓝方 QT 目标进行打击，力争将蓝方部队击溃。

UUV 上共有 8 条 YL。

目标位于红方东侧 60 海里处、对海探测能力范围外。

1.1.2.2　蓝方作战目标

QT 编队自由航行。

蓝方 QT 上共有 8 条 YL，8 条 DD。

目标位于 QT 西侧 60 海里处、对海探测范围外。

1.1.2.3　想定分析

双方舰队距离 60 海里内，以现代武器装备的探测能力，双方已处于遭遇状态，彼此先后进入对方的探测、打击范围。红方通过接口接入指挥智能体进行战斗训练，蓝方使用现有战术条例进行战斗。战斗的胜负取决于战场指挥决策能力。

1.2　指挥智能体设计

根据想定设计，智能体训练平台可解析智能体的基本状态，根据智能体的观测状态空间和数据类型以及动作空间和数据类型。

初始化生成红方“智能体”对象，其初始机器训练学习规则为“随机”；然后设置“蓝方”战术规则，并设置仿真系统作战规则，两方在该设置下可以具备训练 N 局(一个轮次)的条件。

判定“智能体”训练结束，具体实现方法：根据“红方”和“蓝方”智能体的胜率分布，如果该分布逐渐收敛达到没有明显胜率提升的状况则停止训练，如红方胜率高，则判定“智能体”训练第一阶段完成；如红方胜率低，则继续以该方式进行多轮次对抗，直到胜率高于蓝方。

训练如表 1 所示。

表 1　训练

训练轮数	指挥智能体	固定规则	红方达到的胜率/%
1	红方	蓝方	XX
2	红方	蓝方	XX
3	红方	蓝方	XX
⋮	⋮	⋮	⋮
n	……	……	XX

1.3　强化学习特征工程

在当前想定下，除仿真环境直接返回的作战地图/红方智能体/蓝方的属性信息，基于已有的属性值构建出包含更多信息量的特征，包括如下特征：

计算智能体对某一目标的威胁度：

威胁度=期望战果(本次攻击得分)×智能体价值+可能的攻击得分(射程覆盖我方智能体的平均分数)。

公共视野：红方所有智能体视野范围内的全部蓝方智能体 id 列表，即公共视野；

蓝方在我方视野范围内的单位数；

蓝方视野范围内蓝方兵力剩余比例；

蓝方视野范围内蓝方兵力距离我方最近智能体的平均距离；

蓝方视野范围内蓝方兵力射程范围内的我方单位占比；

蓝方公共射程范围内蓝方兵力剩余比例；

公共范围内蓝方兵力距离我方最近智能体的平均距离；

我方领先的分数；

视野内蓝方兵力对我方智能体的最高威胁度、平均威胁度、总威胁度；

我方智能体对视野内蓝方兵力的最高威胁度、平均威胁度、总威胁度；

敌我双方的威胁度之差；

敌我双方的威胁度之比；

蓝方本方所有智能体与蓝方兵力距离的平均值；

本方所有智能体与蓝方兵力距离的最大值；

本方所有智能体与蓝方兵力距离的最小值。

特征工程可以随着训练的进行不断优化，从而加速训练模型收敛。

2　指挥智能体学习训练

指挥智能体的学习训练分为想定筹加载、接口设计、状态设计、数据适配、决策解析设计、决策规则库设计、算法设计和优化、指挥智能体应用与效果 8 阶段。

2.1　想定加载

根据想定设计，完成仿真系统的想定配置，生成想定文件，根据不同的想定，预先生成想定文件。

2.2　接口设计

指挥智能体接口设计为：

Environment.agents：环境中智能体的相关信息；

Envrionment.num_agents：环境中智能体的数量；

Environment.is_terminate：交互是否结束；

Environment.initialize：环境初始化；

Environment.step：智能体与环境交互；

Environemnt.get_current_agents：返回当前行动的智能体列表；

Environment.gen_observation：生成观测类。

2.3 状态设计

状态设计模块通过对原始观测状态数据进行分析，采用大量函数重新设计，构造出更细致的观测状态。其主要功能如下。

2.3.1 调用专家经验规则库

设计出重要观测状态属性的计算公式，如：“威胁度”计算公式、武器伤害修正公式等。

2.3.2 生成观测状态的全局态势

通过获取原始观测状态以及专家经验规则库中内置的先验知识，提取全局状态信息和全局统计信息。全局状态信息是智能体视角下整个战场的状态，如推演时间进程、双方地理态势、双方推演得分。全局统计信息是智能体视角下敌我双方智能体的基本统计信息，如单位数量、总兵力、剩余兵力占比和总威胁度评估等。

2.3.3 生成观测状态的局部态势

根据专家经验规则库中内置的先验知识，在原始观测状态上，扩充每个智能体自身的属性，包括智能体类型、机动状态、油耗状态、所携带武器类型、剩余弹药数量、武器冷却时间、所处高程、行进速度和威胁度等[5-7]。

2.4 数据适配

将观测状态按照观测状态模版生成强化学习平台需要的标准输入格式，然后将解析后的低级动作指令按照动作空间模板转化成兵棋推演系统需要的标准输入格式。

2.5 决策解析设计

2.5.1 生成局部对战规则动作指令

生成局部对战规则动作指令。将强化学习平台返回的高层决策，如拉升、射击、巡逻、返航、静止等根据专家经验规则库的内置局部作战规则动作指令。

2.5.2 生成作战计划动作指令

如果高层决策为执行预知作战计划，则按照决策规则库中预先设定的规则顺序执行一系列低级动作指令。

2.6 决策规则库设计

决策规则库的主要功能包括为状态设计模块提供先验知识、支持将宏观决策解析成协同作战的经验和为决策解析模块提供某些场景下的作战计划 3 部分，状态设计模块提供先验知识、通视规则、期望战斗结果估算等规则，支持对原始观测状态进行计算，并支持将宏观决策解析成协同作战的经验。

2.7 算法设计与调优

分析、研究业务和想定特点，指挥智能体选择了近端策略优化(PPO)算法作为基础算法，PPO 算法改进了目标函数，使用随机梯度法就能够更新策略模型，同时为实现更好的效果，PPO 算法在策略模型目标函数上作了限制，并且在价值模型使用 GAE 算法进行优势函数估计。

2.7.1 处理连续动作

PPO 算法既能处离散动作，又能处理连续动作。在离散动作场景下，动作是可数的，离散动作 0 表示向左走，离散动作 1 表示向右走，每个动作都有对应的概率值，随机进行抽取，获取一个随机性策略，而在连续动作场景中，输入的是某一个浮点类型的动作，在 DQN 算法中，状态行为价值函数 Q 本质上是一个 Q-table，将状态下选择的每个动作所产生的价值都分布到一张大表中。把连续型动作也进行类似处理：将连续型概率切分成多个小份，每个小份的数值代表动作的概率，这其实是一种穷举的思想。在连续型动作中，使用函数来代替数组，如策略分布函数 P=(action)表示在该策略下选择某一行为动作的概率为 P。

在引入深度神经网络时，网络输出的策略为一个固定格式，并不是连续的；因此，可以用分布的方式去表示连续型的概率。先假设策略分布函数服从一个可以用有限个参数表示的特殊分布，如正态分布可以用 μ，σ 这 2 个参数来表示：μ 是分布图像的中心轴，表示平均值，影响图像左右移动；σ 表示方差，σ 越大图像越扁平，σ 越小图像越高瘦。这样神经网络的输出是 2 个参数，根据这 2 个参数，即可获得该策略的概率密度函数。只要按照概率选取一个动作，在整个概率密度函数中抽样出来即可。

2.7.2 重要性采样

PPO 算法是 Off-Policy 的，将目标策略和行为策略分开，智能体已有的是目标策略，但是不针对这个策略进行采样，而在行为策略上进行采样和更

新价值函数。行为策略可以是复用先前学习到的策略，也可是模仿学习人类策略等一些较为优化成熟的策略。这样可在保持探索的同时，更能求到全局最优值。如在智能体与环境交互时为交互数据做上版本标记，在使用新版本数据进行策略优化时，老版本数据同样可用。

需要注意的是，On-Policy 只在一个策略上进行采样，不能使用其他策略。假设在状态 S 下可以选择 a_1 和 a_2 2 个动作，现有策略 π：选择 2 种动作的概率都是 0.5；策略 μ：0.1 的概率选择 a_1，0.9 的概率选择 a_2。那么在策略 μ 上进行采样得到的数据，是不可以用来更新策略 π 的。这是因为在 On-Policy 如策略梯度算法中，会将 TD 误差作为权重去更新策略，更新幅度与权重大小成正比。策略 μ 所产出的数据中动作 a_1 的 TD 误差比动作 a_2 要大，但出现的概率远小于 a_2，因此会是策略 π 的中动作 a_2 的概率也变大了。而在 Off-Policy 算法，如 DQN 则可以多次重复使用数据，这是因为 DQN 并不是优化策略，而是去更新 Q 值，在某一动作后，可能会达到不同的状态，但是概率并不是由策略决定的，而是从环境中得到的，所以所产生的数据和策略不相关。并且在 DQN 是“有目标”的更新，会向目标靠近并逐渐收敛；而策略梯度算法是不断远离原来的策略分布，远离多少以及远离次数的比例需要把控好。

PPO 算法是通过重要性采样来实现离线更新策略，要想用策略 μ 来更新策略 π，需要给 TD 误差乘以一个重要性权重。重要性权重：

$$IW=\pi(a)/\mu(a)。\tag{1}$$

也就是说在 PPO 算法中，用行为 a 在目标策略中出现的概率除以这个动作出现在行为策略中出现的概率。以上面的策略 π 和 μ 为例，可计算出如表 2 所示策略表。

表 2　策略

动作	策略 π	策略 μ	重要性权重	TD 误差	带权重的 TD 误差
a_1	0.5	0.1	5.00	1.5	7.50
a_2	0.5	0.9	0.56	1.0	0.56

在将 TD 误差乘以重要性权重后，a_1 的 TD 误差大幅度提升，a_2 误差减少，即使用策略 μ 去更新策略 π，动作 a_1 提升的概率会变大。重要性采样的公式表示如下：

$$E_{x\sim\pi}[f(x)]=E_{x\sim\mu}\left[f(x)\frac{\pi(x)}{\mu(x)}\right]。\tag{2}$$

需要注意的是，重要性采样所使用的行为策略分布 $\pi(x)$与目标策略分布 $\mu(x)$要求相差不大如果二者分布相差较远，就需要多次的采样才能得到近似的结果，如图 1 所示。

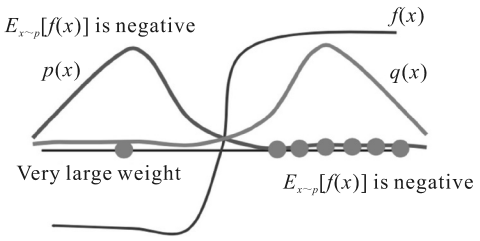

图 1　重要性采样

很显然，在 x 服从分布 $p(x)$时，$f(x)$的期望为负，此时从 $q(x)$中来采样少数的 x，那么采样到的 x 很有可能都分布在右半部分，此时 $f(x)>0$，很容易得到 $f(x)$期望为正的结论，这就会出现问题，因此需要进行大量的采样。从上图可以看出，实际上只是做了一个技巧，通过 2 个采样的概率密度比值来明确分布 $q(x)$在某点采样之余分布 $p(x)$在某点采样的参考程度，将 $f(x)$对于分布 $p(x)$的期望值转化为相对于另一个分布 $q(x)$的期望值，即在已知分布 $q(x)$各点采样值时，通过各采样点的重要性权重 $p(x)/q(x)$可以还原出分布 $p(x)$在该点的采样值；因此，可以将策略 q 下的采样数据用于策略 p 的训练。

在 PPO 算法中应用了重要性采样方法，可使用行为策略获取的数据更新目标策略，从而将 Actor-Critic 框架从在线策略转换成离散策略。得到目标函数为：

$$J_{\pi'\theta}(\theta)=\frac{\nabla J_{\pi'\theta}(\theta)}{\nabla\log\pi_\theta(\alpha|s)}\approx E\left[\frac{\pi_\theta(\alpha|s)}{\pi'_{\theta'}(\alpha|s)}A(s,a)\right]。\tag{3}$$

由于以上近似的原因，实际使用的异步策略与原策略之间的差距不应过大，否则会导致方法不适用。因此，对于距离进行约束后便可得到 PPO 算法。

2.7.3　算法主要流程

PPO 算法除了对连续型动作的输出进行处理并引入重要性采样，还使用 n-step 更新技术，对 TD(0)算法进行引申得到 TD(n)算法[8-9]。实际上只要计算最后的状态价值 $V(s')$，根据这个估算的 $V(s')$就可以反推所有经历过状态的价值。与策略梯度算法中对收获值进行估算不同的是，并不需要经历完整的 episode，而是在中途进行终端用神经网络进行估算。

在 PPO 算法中，使用参数 N 表示并行运行的 Actor 格式，并设置截断长度 T，每次对网络参数进行更新的过程中，所有的 Actor 共产生 NT 各数据。对这些数据进行 epochs 为 K，batchsize 为 M 的网络

更新。其算法为代码：

Algorithm PPO, Actor−Critic Style

for iteration=1, 2, …, do

for actor=1, 2, …, N do

Run policy $\pi_{\theta\text{-}old}$ in environment for T timesteps

Compute advantage estimates A_1, …, A_t

end for

Optimize surrogate L wet θ, with K epochs and minibatch size $M \leqslant NT$

θ−old $\leftarrow$ θ

end for

2.8 指挥智能体应用与效果

经上述学习训练的 3 个指挥智能体实装 UUV 演示反 Q 场景如图 2—5 所示，其中蓝方为传统方式定义的战术战役规则。图 2 为在疑似目标方向多个指挥智能体 UUV 在同一方向上进行接替方式探测执行任务；图 3 为在疑似目标方向采用多条线路并行进行探测执行任务。然后，进行协同打击(图 4)，牺牲了 1 条 UUV，打击蓝方胜利后，剩余的 2 条 UUV 继续回到原路线进行后续任务(图 5)。

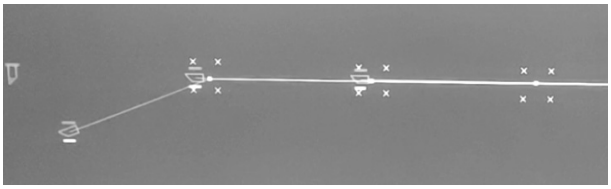

图 2 多条 UUV 协同接替探测目标

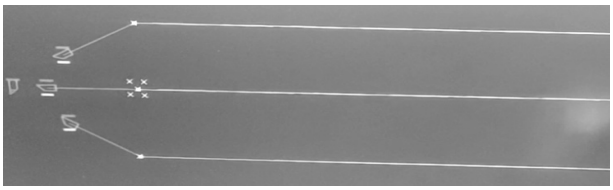

图 3 多条 UUV 并行探测目标

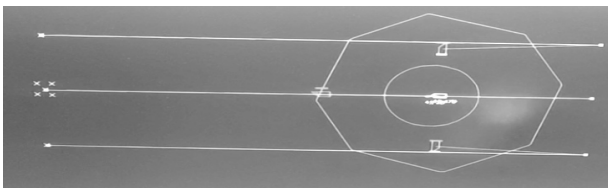

图 4 多条 UUV 进行协同打击目标，其中一条 UUV 被击中

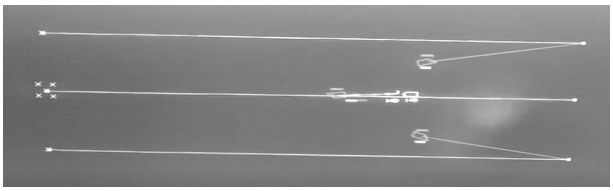

图 5 指挥体智能体的红方获胜后继续执行后续任务

上述训练好的指挥智能体应用效果表明，基于深度强化学习的训练、得到充分考虑环境变化下执行最优决策的指挥智能体，通过数据驱动的方式取代传统基于规则的方式，有效解决了弱通信约束条件下 UUV 的任务分配、航路规划、指挥控制问题。

3 结束语

笔者围绕 UUV 的任务分配、航路规划、指挥控制问题，提出一种基于 PPO 强化学习训练指挥智能体的方案。通过假定作战场景，转换成符合战术规则的想定，并对指挥智能体进行建模，从而构建出指挥智能体的训练系统。通过该系统设计和优化 PPO 强化学习算法，不断提高红方指挥智能体的胜率并趋于稳定。该指挥智能体应用效果表明：通过强化学习训练指挥智能体能够实现无人水下航行器的任务规划、航路规划、指挥控制并胜于基于规则的传统方式。不断优化强化学习算法，可以不断提高指挥智能体的胜率，从而为无人水下航行器的任务分配、航路规划、指挥控制的研究发展提供一种尝试。

参考文献：

[1] 孙现有, 马琪. 美海军 UUV 使命任务必要性与技术可行性分析[J]. 鱼雷技术, 2010, 18(3): 231−235.

[2] 徐卓. 基于神经网络算法的无人机航迹规划研究[D]. 石家庄: 河北科技大学, 2016.

[3] CHEN B, LIN C, LIU X P, et al. Adaptive fuzzy tracking control for a class of MIMO nonlinear systems in non-strict-feedback form[J]. IEEE Transactions on Cyber-netics, 2014, 45(12): 2743−2755.

[4] 李聪, 贾红军. 无人水下航行器的智能航行控制[J]. 舰船科学技术, 2018, 40(4A): 3−6.

[5] 张玉平, 王有成, 赵铜星, 等, 区间直觉模糊决策在联合作战指挥员能力评估中的应用[J]. 兵工自动化, 2013, 32(11): 45−48.

[6] PRESTERO T. Verification of a six degree of freedom simulation model for the REMUS autonomous under atervehicles[D]. USA: MIT, 2001.

[7] 方兴. 基于贝叶斯网络的水下目标识别[J]. 舰船电子工程, 2020, 40(9): 41−43, 61.

[8] 葛峰, 韩建立, 高松. 基于 BAS-BP 神经网络的多应力加速寿命试验预测方法[J]. 兵工自动化, 2020, 39(6): 5−9.

[9] 苏玉民, 曹健, 徐峰, 等. 鱼雷型水下机器人非线性航迹跟踪控制[J]. 上海交通大学学报(自然科学版), 2012, 46(6): 977−983.