

近水面协同对抗博弈模型的构建与实现

邹修玮¹
ZOU Xiuwei

摘要

自主水下航行器 (autonomous underwater vehicle, AUV) 凭借其具备的自主导航、隐蔽性强、可长时间在复杂水下环境执行任务等特点, 在水下对抗中扮演着关键角色, 已成为各国提升水下作战能力的重要装备。在水下多 AUV 对抗场景中, 目标分配问题是决定作战效能的核心因素。合理的目标分配能够优化 AUV 的资源利用, 充分发挥其作战优势, 提高任务完成的成功率。反之, 不合理的目标分配可能导致 AUV 资源浪费、任务冲突, 甚至无法达成作战目标。基于此, 文章构建了基于动态博弈理论的数学模型, 并设计了纳什均衡值求解算法。通过引入混合策略应对不确定性, 适应度函数将求解转化为优化问题。此外, 采用粒子群优化算法迭代搜索最优混合策略, 仿真结果表明: 所提算法能有效找到纳什均衡解, 为制定对抗策略提供了科学依据。

关键词

水下多 AUV 对抗; 纳什均衡; 粒子群优化算法

doi: 10.3969/j.issn.1672-9528.2025.10.016

0 引言

海洋作为连接世界各国的重要纽带, 其战略意义日益凸显, 已成为各国维护国家主权、保障经济发展和提升国际影响力的关键领域。海洋资源的开发利用、海上贸易通道的安全维护以及海洋权益的保护, 都对国家的发展和国家安全至

重要。随着海洋战略地位的提升, 水下作战领域的重要性也愈发显著。自主水下航行器 (autonomous underwater vehicle, AUV) 凭借其具备的自主导航、隐蔽性强、可长时间在复杂水下环境执行任务等特点, 在水下对抗中扮演着关键角色, 已成为各国提升水下作战能力的重要装备^[1]。

在水下多 AUV 对抗场景中, 目标分配问题是决定作战效能的核心因素。合理的目标分配能够优化 AUV 的资源利

1. 扬州大学物理科学与技术学院 江苏扬州 225002

参考文献:

- [1] 周晨曦, 赵天驰, 张玲玲. 基于用户画像和链路预测的电影个性化推荐研究 [J]. 系统科学与数学, 2024, 44(6): 1586-1607.
- [2] 王新武. 基于隐马尔可夫模型的在线视频个性化推荐研究 [J]. 数字通信世界, 2023(9): 1-4.
- [3] 刘瑞, 陈烨. 考虑长短期兴趣及其演化的电影个性化动态推荐研究 [J]. 数据分析与知识发现, 2024, 8(1): 80-89.
- [4] 魏子钦, 单豫洲, 梁艳美. 基于混合推荐算法的电影推荐研究 [J]. 信息与电脑 (理论版), 2022, 34(9): 166-168.
- [5] 李彭. 基于用户特征与信任度的协同过滤推荐算法 [J]. 现代信息科技, 2024, 8(24): 49-53.
- [6] 王锐, 王永, 刘金源, 等. 基于 Rényi 差分隐私的图卷积协同过滤推荐算法 [J]. 软件学报, 2025, 36(3): 1202-1217.
- [7] 王永, 李行健, 邓江洲. 融合注意力机制的残差神经网络协同过滤推荐模型 [J]. 运筹与管理, 2024, 33(10): 201-208.

- [8] 王宁, 李然, 王客程, 等. 基于神经网络和注意力机制的协同过滤推荐算法的研究 [J]. 现代电子技术, 2024, 47(20): 95-100.
- [9] 徐远宏. 基于协同过滤的农业平台学习资源个性化推荐方法探究 [J]. 科技资讯, 2024, 22(16): 174-176.
- [10] 崔春生, 魏盟, 车利斌, 等. 基于概率语言术语集的非个性化推荐算法研究 [J]. 系统科学与数学, 2023, 43(11): 2990-3010.
- [11] 景楠, 周正茜, 袁戟. 基于宽度学习和协同过滤的在线信息推荐算法 [J]. 计算机应用与软件, 2023, 40(5): 279-287.

【作者简介】

时小芳 (1991—), 女, 河南濮阳人, 硕士研究生, 讲师、教研室主任, 研究方向: 计算机视觉、机器学习。

向志华 (1982—), 女, 湖南邵阳人, 硕士, 副教授、教研室主任, 研究方向: 软件工程、机器学习。

(收稿日期: 2025-04-23 修回日期: 2025-10-14)

用,充分发挥其作战优势,提高任务完成的成功率。反之,不合理的目标分配可能导致 AUV 资源浪费、任务冲突,甚至无法达成作战目标^[2]。例如,在反潜作战中,若能精准地将 AUV 分配到目标潜艇可能出现的区域,就能极大地提高搜索和攻击的效率,增强作战的主动权。因此,研究水下多 AUV 对抗中的目标分配问题,对于提升水下作战效能、维护国家海洋安全具有重要的现实意义。本文主要研究水下多自主水下航行器(AUV)对抗中的目标分配问题,着力构建基于动态博弈理论的水下目标分配导向模型,并设计了基于粒子群优化算法(PSO)的纳什均衡值求解算法,为制定对抗策略提供了科学依据。

1 近水面多智能体动态博弈建模与纳什均衡值求解

1.1 水下目标分配导向的动态博弈模型构建

本文构建了一个非合作对抗策略问题的数学模型 $G = \langle I, S, U \rangle$ 。在此模型中,局中人集合 I 由 A 方和 D 方组成,代表对抗的双方($I = \{A, D\}$)。策略集合 S 则包含 A 方和 D 方各自可能采取的所有纯策略,这些纯策略构成了双方进行博弈的基础行为单元($S = \{S^A, S^D\}$)。其中, A 和 D 纯策略个数分别为 m 和 n ,二者纯策略集分别为: $S^A = \{S^A_1, S^A_2, \dots, S^A_m\}$; $S^D = \{S^D_1, S^D_2, \dots, S^D_n\}$ 。

A 的混合策略是 $X = (x_1, x_2, \dots, x_i)$, x_i 是 A 选择第 i 个纯策略的概率; D 的混合策略是 $Y = (y_1, y_2, \dots, y_j)$, y_j 是 D 选择第 j 个纯策略的概率。此外,还需要满足条件:

$$\sum_{i=1}^m x_i = 1, \sum_{j=1}^n y_j = 1 \quad (1)$$

收益矩阵 $U = \{U^A, U^D\}$, A 方和 D 方的收益矩阵 U^A 和 U^D 分别记录了双方在不同策略组合下的生存概率值。其中 U^A_{ij} 是 A 选择第 i 个纯策略且 D 选择第 j 个纯策略时 A 的收益; U^D_{ij} 是第 i 个纯策略且 A 选择第 j 个纯策略时 D 的收益。

双矩阵博弈中,纳什均衡解(X, Y)的充要条件是: A 方用混合策略 X 、 D 方用混合策略 Y 时,双方都无法通过单独改用任何一个纯策略来提高自身收益。即对于所有局中人 A 的纯策略 S^A_i 和局中人 D 的纯策略 S^D_j ,混合策略 X 和 Y 须满足:

$$U^A_{i\cdot} Y^{*T} \leq X^* U^A Y^{*T}, X \cdot U^D_{\cdot j} \leq X \cdot U^D Y^{*T} \quad (2)$$

式中: $i = 1, 2, \dots, m$; $j = 1, 2, \dots, n$; $U^A_{i\cdot}$ 是矩阵 U^A 的第 i 行; $U^D_{\cdot j}$ 是矩阵 U^D 的第 j 列。

为求解纳什均衡点,本文设计了适应度函数 $f(x, y)$ 。适应度函数定义为:

$$f(x, y) = \{\max(U^A_{i\cdot} Y^{*T} - X^* U^A Y^{*T}), 0\} + \{\max(XD_j - X^* U^D Y^{*T}), 0\} \quad (3)$$

1.2 基于粒子群算法的博弈对抗纳什均衡值求解算法设计

1.2.1 算法设计与描述

初始化阶段,算法随机生成一定数量的粒子,并赋予其

初始的位置和速度。假设 A 方有 n_A 种策略, D 方有 n_D 种策略,则一个粒子的编码可以表示为:

$$X_i = (X_{i1}, X_{i2}, \dots, X_{in_A}, Y_{i1}, Y_{i2}, \dots, Y_{in_D}) \quad (4)$$

式中: $X_{i1}, X_{i2}, \dots, X_{in_A}$ 为 A 方的混合策略,满足 $\sum_{j=1}^{n_A} X_{ij} = 1$ 且 $X_{ij} \geq 0$; 同理, $Y_{i1}, Y_{i2}, \dots, Y_{in_D}$ 为 D 方的混合策略,满足 $\sum_{j=1}^{n_D} Y_{ij} = 1$ 且 $Y_{ij} \geq 0$ 。在初始化阶段,笔者随机生成每个粒子的位置和速度,确保满足上述约束条件。

在混合策略优化中,由于策略概率必须满足特定的约束条件,因此速度和位置的更新不能简单地按照原始粒子群优化算法公式进行。速度和位置的更新公式为:

$$V_{ij}^{k+1} = \omega V_{ij}^k + c_1 r_1 (P_{ij}^{\text{best}} - X_{ij}^k) + c_2 r_2 (G_{ij}^{\text{best}} - X_{ij}^k) \quad (5)$$

式中: V_{ij}^{k+1} 是第 i 个粒子在第 $k+1$ 次迭代中第 j 维的速度(对于 A 方或 D 方的某个策略); ω 是惯性权重; c_1 和 c_2 是学习因子; r_1 和 r_2 是随机数; P_{ij}^{best} 是该粒子自身历史最优位置的第 j 维; G_{ij}^{best} 是整个种群历史最优位置的第 j 维。

位置更新为:

$$X_{ij}^{k+1} = X_{ij}^k + V_{ij}^{k+1} \quad (6)$$

同样,位置更新后需要确保满足混合策略的约束条件。

1.2.2 算法实现

(1) 初始化阶段

算法的初始化是求解过程的第一步,旨在生成一组随机分布的粒子,每个粒子包含两个主要的变量部分,分别用 $x(i, 1:16)$ 和 $x(i, 17:32)$ 表示,这两部分共同构成了粒子的 32 维位置向量。其中, $x(i, 1:16)$ 代表粒子的前 16 个变量的位置,而 $x(i, 17:32)$ 则代表后 16 个变量的位置。通过 rand 函数为这些位置生成随机数,随后进行归一化处理。

(2) 适应度计算与全局最优更新

初始化后,算法进入适应度计算与全局最优更新阶段。本部分借助 for 循环遍历粒子,调用 fitness 函数,输入粒子位置信息($x(i, 1:16)$ 、 $x(i, 17:32)$)及可能的 U^A 、 U^D 参数,计算适应度值存于数组 p ,粒子当前位置存于矩阵 y ,方便后续更新位置、搜索最优解,助力算法逼近全局最优解。

(3) 全局最优位置初始化

本部分先将全局最优位置 p_g 初始化为最后一个粒子位置,即从代表粒子位置矩阵 x 中取最后一行 $x(N)$ 赋值给 p_g 。接着,用 for 循环遍历除最后一个粒子外的其他粒子,每次迭代时用 fitness 函数算出当前 p_g 适应度,并与遍历到的粒子适应度 $p(i)$ 对比,若 $p(i)$ 更优,就把 p_g 更新为当前粒子位置。

(4) 迭代更新过程

本部分执行 M 次迭代,每次对所有粒子的速度和位置进行更新。每次迭代时,先依据粒子群算法公式更新粒子速度,公式纳入惯性权重 ω (决定粒子当前速度对下一时刻速度的

影响程度)、个体学习因子 c_1 (反映粒子对自身历史最优位置的信任程度) 以及社会学习因子 c_2 (体现粒子对群体历史最优位置的信任程度), 借此平衡自身与群体经验, 实现合理的速度更新。随后, 对速度加以约束, 防止因速度更新致使粒子位置变化违反特定约束条件 (如位置不能小于 0), 确保位置更新有效。

在速度和位置更新之后, 代码对粒子的位置进行了重新归一化。然后, 代码计算了更新后粒子的适应度, 并根据适应度值更新了每个粒子的个体最优位置及其对应的适应度值。最后, 代码还更新了全局最优位置。

(5) 结果展示

首先, “ $x_m=p_g'$;” 这行代码将全局最优位置 p_g 转置并赋值给 x_m , 以便以列向量的形式表示全局最优解的位置。接着执行命令: “ $f_v=\text{fitness}(p_g(1:16), p_g(17:32), U^d, U^p)$;” 这部分计算了全局最优位置 p_g 对应的适应度值, 并将其赋值给 f_v 。

2 近水面协同对抗仿真与验证

2.1 仿真场景设计

本文设定了各攻击单元在理想情况下的攻击毁伤概率, 分别为:

$$\begin{aligned} P_{1,1}^A=0.9、P_{1,2}^A=0.8、P_{2,1}^A=0.25、P_{2,2}^A=0.15 \\ P_{1,1}^D=0.4、P_{1,2}^D=0.75、P_{2,1}^D=0.5、P_{2,2}^D=0.8 \end{aligned} \tag{7}$$

A 方第一个攻击单元对 D 方第二个目标的攻击概率为 80%($P_{1,2}^A=0.8$), 毁伤概率为 75%($P_{1,2}^D=0.75$)。

A 方第二个攻击单元对 D 方第一个目标的攻击概率为 25%($P_{2,1}^A=0.25$), 毁伤概率为 50%($P_{2,1}^D=0.5$)。

A 方第二个攻击单元对 D 方第二个目标的攻击概率为 15%($P_{2,2}^A=0.15$), 毁伤概率为 80%($P_{2,2}^D=0.8$)。

这些概率反映了不同 AUV 和武器组合在攻击时的毁伤能力。同时, 取环境影响因子 γ 为 1, 即不考虑环境因素对攻击效果的影响。

本部分执行 M 次迭代, 每次迭代都会更新全部粒子的速度和位置。更新速度时, 运用粒子群算法公式, 该公式包含惯性权重 w 、个体学

习因子 c_1 和社会学习因子 c_2 , 以此平衡自身与群体经验。之后对速度进行约束, 避免速度更新使粒子位置违反诸如位置不能小于 0 等特定条件, 保证位置更新有效。如表 1 所示。

表 1 A 方攻击单元对 D 方目标的攻击效能扩展表

单位: %

攻击单元	目标	AUV 机动性	攻击概率 (α)	毁伤概率 (β)	综合毁伤率 ($\alpha \times \beta$)
A1	目标 D1	高	90	40	36
A1	目标 D2	中	80	75	60
A2	目标 D1	低	25	50	12.5
A2	目标 D2	高	15	80	12

2.2 智能体收益矩阵计算

根据 A、D 双方可能采取的策略组合, 计算了 A 方的收益, 并形成了收益矩阵。矩阵中的每一个元素都代表了 A 方和 D 方选定一个策略组合后, A 方的收益。然后, 计算了在 D 方采取不同策略时, A 方采取每一种策略的收益。这些收益值就构成了 A 方的收益矩阵。通过计算收益矩阵, 可以清晰地看到在不同策略组合下, A 方的收益情况。如表 2 所示。表 2 展现了 A、D 双方经过两个战斗步骤后的策略选择对 A 方生存概率的影响, 行代表 A 方策略编号及具体策略, 列展示 D 方策略, 表中数值精确反映特定策略组合下 A 方生存概率, 其高低体现 A 方策略优劣, 行列对比可评估不同策略影响 [3]。

表 2 K=2 时 A 方的收益矩阵

		A 方策略											
D 方策略		1	2	3	...	7	8	9	...	13	14	15	16
		1,1:1,1	1,1:1,2	1,1:2,1	...	1,2:2,1	1,2:2,2	2,1:1,1	...	2,2:1,1	2,2:1,2	2,2:2,1	2,2:2,2
1	1,1:1,1	1.142	1.140	0.532	...	1.165	1.161	1.189	...	1.521	1.518	1.524	1.520
2	1,1:1,2	0.491	0.487	0.494	...	0.625	0.627	0.048	...	1.424	1.014	1.023	1.021
3	1,1:2,1	1.090	1.093	1.097	...	1.097	1.093	0.735	...	0.512	0.526	0.517	0.525
4	1,1:2,2	0.485	0.487	0.483	...	0.596	0.597	0.663	...	0.517	0.554	0.582	0.547
5	1,2:1,1	0.482	0.492	0.485	...	0.521	0.523	0.574	...	0.519	0.525	0.572	0.519
6	1,2:1,2	0.630	0.625	0.627	...	0.623	0.614	0.547	...	0.543	0.554	0.553	0.513
7	1,2:2,1	0.485	0.481	0.487	...	0.542	0.523	0.617	...	0.528	0.519	0.560	0.528
8	1,2:2,2	0.625	0.647	0.638	...	0.647	0.654	0.682	...	0.663	0.658	0.682	0.563
9	2,1:1,1	0.492	0.497	0.482	...	0.519	0.525	0.572	...	0.591	0.527	0.532	0.591
10	2,1:1,2	0.535	0.534	0.532	...	0.613	0.554	0.553	...	0.516	0.527	0.511	0.526
11	2,1:2,1	0.485	0.487	0.484	...	0.528	0.519	0.560	...	0.666	0.697	0.623	0.696
12	2,1:2,2	0.548	0.543	0.544	...	0.563	0.558	0.582	...	0.926	0.927	0.963	0.926
13	2,2:1,1	0.537	0.536	0.534	...	0.601	0.617	0.692	...	0.521	0.523	0.574	0.521
14	2,2:1,2	0.981	0.982	0.983	...	0.996	0.997	0.911	...	0.644	0.623	0.628	0.645
15	2,2:2,1	0.541	0.543	0.545	...	0.596	0.597	0.663	...	0.926	0.927	0.963	0.926
16	2,2:2,2	1.012	1.014	1.003	...	0.596	0.597	0.663	...	1.021	1.018	1.024	1.020

2.3 基于粒子群算法的纳什均衡求解

由表 3 可知, D 方单方面改变策略时, A 方的收益始终低于 0.778; 同样, 当 A 方保持策略 X 不变而 D 方单方面改变策略时, D 方的收益始终低于 0.812。

表 3 基于 PSO 算法的纳什均衡解

策略编号	策略	A 方最优 X	D 方最优 Y
1	1.1:1.1	1	0
2	1.1:1.2	0	0
3	1.1:2.1	0	0
4	1.1:2.2	0	0
5	1.2:1.1	0	0
6	1.2:1.2	0	0
7	1.2:2.1	0	0
8	1.2:2.2	0	0
9	2.1:1.1	0	0
10	2.1:1.2	0	0
11	2.1:2.1	0	0
12	2.1:2.2	0	0
13	2.2:1.1	0	0
14	2.2:1.2	0	0
15	2.2:2.1	0	0
16	2.2:2.2	0	1

通过对比发现, A 、 D 双方单方面改变策略时, 收益均小于各自在纳什均衡解下的收益, 满足纳什均衡条件, 证明找到的混合策略是纳什均衡解。表 3 列有 16 种策略组合, 呈现了策略编号与具体策略的对应关系, 及 A 、 D 双方的最优决策。 A 方最优选择 (X) 有规律, 前缀为 1.1 或 1.2 时, X 为 1; 前缀为 2.1 或 2.2 时, X 为 2; D 方最优选择 (Y) 较复杂, 多数策略组合下 Y 为 0, 仅在策略编号 16 “2.2:2.2” 时, Y 为 1, 表明特定策略组合下 D 方决策会变化。

从策略均衡与优势策略看, A 方选 2.1 或 2.2 优于 1.1 或 1.2; D 方多数情况选 1.1 或 1.2 有利, 但 A 方选 2.2 时, D 方选 2.2 能获得最优结果。

为验证 (X , Y) 是否为纳什均衡点, 对双方随机生成 200 组混合策略并计算收益, 发现 D 方策略不变时 A 方变策略无法提高收益^[4]。

综上, 混合策略 (X , Y) 是博弈的纳什均衡点, 此时 A 、 D 双方收益分别为 0.778 和 0.812, 验证了算法和策略的有效性, 为理解纳什均衡提供实证支持, 如图 1~2 所示。本文通过绘制适应度函数曲线和策略组合表, 分析了 PSO 算法在寻找博弈论中纳什均衡解的过程, 不仅评估了算法的收敛性和稳定性, 还验证了所找到的纳什均衡解的有效性。这一结果为算法和策略提供了有力的实证支持, 也为深入理解博弈论中的纳什均衡提供了有益的启示。

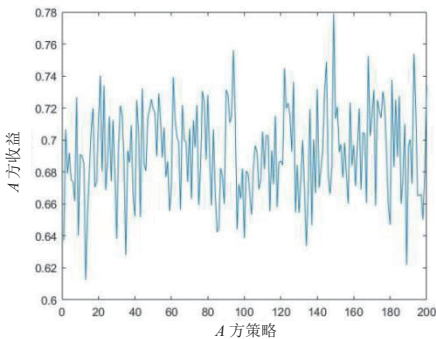


图 1 A 方随机改变混合策略收益

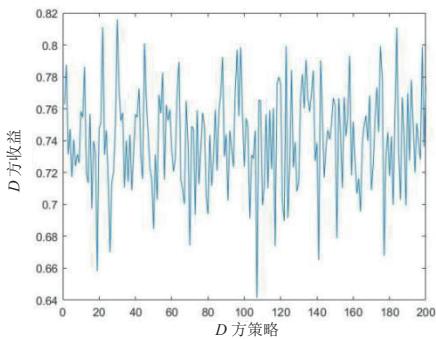


图 2 D 方随机改变混合策略收益

3 结语

本文围绕水下多 AUV 对抗中的目标分配问题, 构建了基于动态博弈理论的水下目标分配导向模型, 并设计了基于粒子群优化算法的纳什均衡值求解算法。通过仿真验证, 结果表明所提算法能够有效找到纳什均衡解, 为水下多 AUV 对抗策略的制定提供了科学依据。

参考文献:

[1] 王辰辰, 王洁, 李婷婷. 无人机集群协同对抗问题的数学建模[J]. 数学的实践与认识, 2021, 51(23): 225-233.

[2] 罗隆, 张邻, 王苏平, 等. 无人机集群协同对抗最优策略研究[J]. 数学的实践与认识, 2021, 51(23): 209-216.

[3] CAO Z J, XU K, JIA H W, et al. An autonomous differential evolution based on reinforcement learning for cooperative countermeasures of unmanned aerial vehicles[J]. Applied soft computing, 2025, 169: 112605.

[4] 顾利东, 李磊, 郝承明, 等. 基于壳外换热的水下航行器壳体型线优化[J]. 科技创新与应用, 2024, 14(29): 8-14.

【作者简介】

邹修玮 (1999—), 男, 江苏淮安人, 硕士研究生, 研究方向: 人工智能。

(收稿日期: 2025-04-03 修回日期: 2025-10-13)