Systematic Review

系统性综述

Scalable and Cooperative Deep Reinforcement Learning Approaches for Multi-UAV Systems: A Systematic Review

多无人机系统可扩展与协同深度强化学习方法: 系统性综述

Francesco Frattolillo *, † (D, Damiano Brunori *,† (D) and Luca Iocchi ( )

Francesco Frattolillo *, † (D, Damiano Brunori *,† (D) 和 Luca Iocchi ( )

Department of Computer, Control and Management Engineering (DIAG), Sapienza University of Rome, 00185 Rome, Italy

计算机控制与管理工程系 (DIAG)，罗马大学，00185 罗马，意大利

* Correspondence: frattolillo@diag.uniroma1.it (F.F); brunori@diag.uniroma1.it (D.B.)

* 联系方式:frattolillo@diag.uniroma1.it (F.F)；brunori@diag.uniroma1.it (D.B.)

† These authors contributed equally to this work.

† 这些作者对本工作的贡献同等。

Abstract: In recent years, the use of multiple unmanned aerial vehicles (UAVs) in various applications has progressively increased thanks to advancements in multi-agent system technology, which enables the accomplishment of complex tasks that require cooperative and coordinated abilities. In this article, multi-UAV applications are grouped into five classes based on their primary task: coverage, adversarial search and game, computational offloading, communication, and target-driven navigation. By employing a systematic review approach, we select the most significant works that use deep reinforcement learning (DRL) techniques for cooperative and scalable multi-UAV systems and discuss their features using extensive and constructive critical reasoning. Finally, we present the most likely and promising research directions by highlighting the limitations of the currently held assumptions and the constraints when dealing with collaborative DRL-based multi-UAV systems. The suggested areas of research can enhance the transfer of knowledge from simulations to real-world environments and can increase the responsiveness and safety of UAV systems.

摘要: 近年来，得益于多代理系统技术的进步，多无人机的使用在各种应用中逐渐增加，这使得能够完成需要协作和协调能力的复杂任务。在本篇文章中，根据主要任务将多无人机应用分为五类: 覆盖，对抗搜索与游戏，计算卸载，通信和目标驱动导航。通过采用系统性综述方法，我们选择了使用深度强化学习 (DRL) 技术实现协同和可扩展多无人机系统的最具代表性的作品，并利用广泛且具有建设性的批判性推理讨论了它们的特点。最后，我们提出了最有可能和最有前景的研究方向，通过突出当前持有假设的限制以及处理基于协同 DRL 的多无人机系统时的约束。建议的研究领域可以增强从模拟到实际环境的知识转移，并可以提高无人机系统的响应性和安全性。

Keywords: unmanned aerial vehicles; multi-UAV; deep reinforcement learning; multi-agent cooperation

关键词: 无人航空器；多无人机；深度强化学习；多代理协作

---

check for updates

检查更新

Citation: Frattolillo, F.; Brunori, D.; Iocchi, L. Scalable and Cooperative Deep Reinforcement Learning Approaches for Multi-UAV Systems: A Systematic Review. Drones 2023, 7, 236. https://doi.org/10.3390/drones7040236

引用:Frattolillo, F.; Brunori, D.; Iocchi, L. 可扩展和协作深度强化学习方法在多无人机系统中的应用: 系统性回顾。Drones 2023, 7, 236. https://doi.org/10.3390/drones7040236

Academic Editor: Marija Popović and Inkyu Sa

学术编辑:Marija Popović 和 Inkyu Sa

Received: 28 February 2023

收到日期:2023 年 2 月 28 日

Revised: 13 March 2023

修订日期:2023 年 3 月 13 日

Accepted: 14 March 2023

接受日期:2023 年 3 月 14 日

Published: 28 March 2023 4.0/).
发表日期:2023 年 3 月 28 日 4.0/).

1. Introduction

1. 引言

Unmanned aerial vehicles (UAVs) have become excellent candidates for addressing various problems due to their high mobility in three-dimensional space, easy deployment, and relatively low production costs. UAVs are utilized in a wide range of applications, from critical humanitarian purposes such as firefighting support [1] and search and rescue (SAR) assistance [2] to purposes aimed at improving the quality of life (QoL) such as parcel delivery (which can decrease delivery costs, see, e.g., [3]), and becoming integrated into industries [4] to monitor the industrial Internet of Things (IoTs) for Industry 4.0. Among the applications of UAVs in communication systems, one example is the use of UAVs to collect data from Internet of Things (IoTs) devices using deep learning (DL) techniques to minimize the Age of Information (AoI) of the data [5]. Another application is to use UAVs to capture views to maximize the fidelity of video reconstruction for remote users [6]. Several applications have been recently developed for scenarios involving surveillance, medical (i.e., blood delivery), agricultural, and high-risk human operations. Many future UAV applications will be based on their cooperative and coordinated interaction, i.e., a fundamental feature of any efficient, robust, and reliable application deployment. Thus, in this article, we decided to examine multi-UAV systems specifically.

无人驾驶飞行器 (UAVs) 由于其三维空间中的高机动性、易于部署以及相对较低的生产成本，已经成为解决各种问题的优秀候选方案。UAVs 在广泛的领域得到应用，从关键的救援任务如消防支援 [1] 和搜救 (SAR) 援助 [2]，到旨在提高生活质量 (QoL) 的用途，例如快递包裹投递 (这可以降低配送成本，例如见 [3])，并且正在融入工业 [4] 中，用于监控工业物联网 (IoTs) 以服务于工业 4.0。在通信系统中 UAVs 的应用之一是使用 UAVs 通过深度学习 (DL) 技术收集物联网 (IoTs) 设备的数据，以最小化数据的时效性 (AoI)[5]。另一个应用是使用 UAVs 捕捉视角，以最大化远程用户视频重建的保真度 [6]。最近还开发了几种涉及监控、医疗 (即血液输送)、农业和高风险人类操作场景的应用。许多未来的 UAV 应用将基于它们协作和协调的互动，即任何高效、健壮和可靠应用部署的基本特征。因此，在本文中，我们决定专门研究多 UAV 系统。

The development of deep reinforcement learning (DRL) and its recent successes have made it an excellent candidate for solving complex problems such as those involving multi-UAV applications. The increase in the computational power of modern machines, along with recent advances in deep learning, has led to remarkable results using DRL, such as playing at a superhuman level.In ATARI 2600 games [7], beating the world champion in the Go strategy board game [8], defeating the world champion in the complex multiplayer game DOTA2 [9], and solving real-life control tasks such as solving a Rubik's cube using a robotic hand under external disturbances [10]. A general overview of the research trends associated with the two main keywords, i.e., reinforcement learning and UAV, is presented in Figure 1. There has been an exponential increase in the number of related publications since 2017.

深度强化学习 (DRL) 的发展及其最近的成就使其成为解决复杂问题的优秀候选方案，例如涉及多无人机 (UAV) 应用的问题。现代计算机计算能力的提升，以及深度学习的最新进展，使得利用 DRL 取得了显著成果，如在国际象棋 ATARI 2600 游戏中达到超人类水平 [7]，在围棋策略棋类游戏中击败世界冠军 [8]，在复杂多人游戏 DOTA2 中战胜世界冠军 [9]，以及解决现实生活中的控制任务，例如在受到外部干扰的情况下使用机械手解魔方 [10]。图 1 展示了与两个主要关键词“强化学习”和“无人机”相关的研究趋势概述。自 2017 年以来，相关论文数量呈指数级增长。

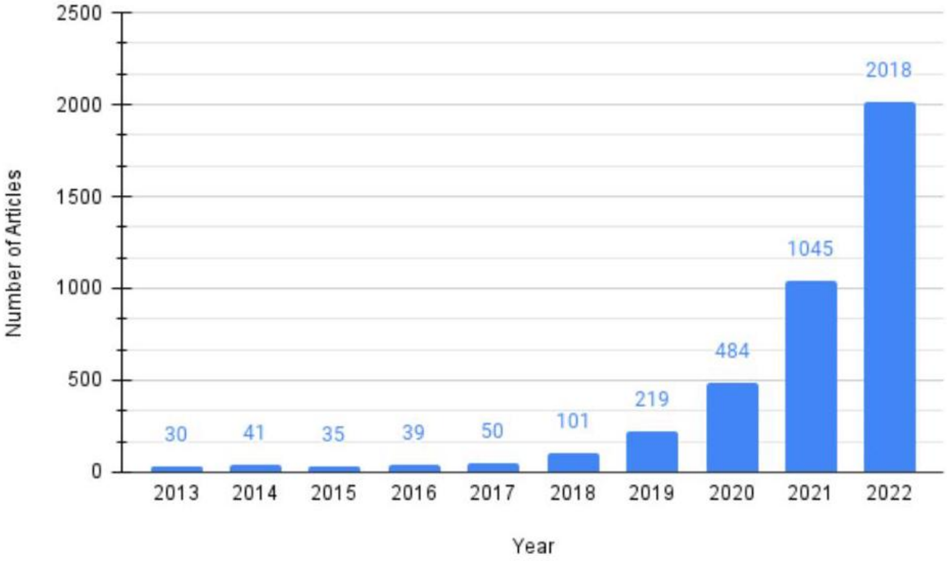


Figure 1. Number of works per year retrieved using the keywords "UAV" and "reinforcement learning".
图 1。使用关键词“无人机”和“强化学习”检索到的每年作品数量。

We then decided to investigate a particular subclass of works, i.e., those involving deep reinforcement learning techniques applied to scalable and collaborative multi-UAV systems. To the best of our knowledge, this article is the first to present an extensive and comprehensive comparative analysis of cooperative DRL-based multi-UAV systems. Our research can guide future studies in this field, which is anticipated to attract increasing attention in the coming years.

然后，我们决定调查一个特定的子类别，即那些涉及将深度强化学习技术应用于可扩展和协作多无人机系统的作品。据我们所知，本文是第一个提出基于协作 DRL 的多无人机系统全面和综合比较分析的。我们的研究可以指导该领域未来的研究，预计未来几年这一领域将吸引越来越多的关注。

Several reviews and surveys of UAVs have already been published but with different perspectives than this article. A more general survey of all the machine learning (ML) techniques applied to UAV systems was provided in the study by Bithas et al. [11], which investigated UAV-based communication applications. A similar topic was covered by Aissa and Letaifa [12], where ML techniques were studied to address some of the constraints of UAV wireless communications. Other works either focus on a specific UAV application or do not focus on DRL techniques. More general AI applications (including RL) for UAV path planning problems were investigated in [13], and specific UAV scenarios, e.g., Internet of Things (IoT) assistance [14], were examined by describing the main approaches used (not only (D)RL-based). It is worth mentioning the study of Azar et al. [15], which focused on investigating DRL techniques related to three main UAV macro-problems, namely path planning, navigation, and control, without specifically addressing collaborative multi-UAV systems and their applications. Their study provided a detailed explanation of some DRL algorithms, which can be useful for better understanding the DRL approaches mentioned in Section 4 of this article.

已经发表了多篇关于无人机的综述和调查，但与本文的视角不同。Bithas 等人 [11] 的研究提供了一个关于所有应用于无人机系统的机器学习 (ML) 技术的更广泛的综述，该研究调查了基于无人机的通信应用。Aissa 和 Letaifa[12] 探讨了类似的主题，其中研究了 ML 技术来解决无人机无线通信的一些限制。其他作品要么专注于特定的无人机应用，要么不专注于深度强化学习 (DRL) 技术。在 [13] 中调查了用于无人机路径规划问题的更广泛的 AI 应用 (包括 RL)，并描述了使用的主要方法 (而不仅仅是基于 (D)RL 的方法) 来研究特定的无人机场景，例如物联网 (IoT) 辅助 [14]。值得提到的是 Azar 等人 [15] 的研究，该研究专注于调查与三个主要无人机宏观问题相关的 DRL 技术，即路径规划、导航和控制，而没有特别解决协同多无人机系统及其应用。他们的研究详细解释了一些 DRL 算法，这有助于更好地理解本文第 4 节中提到的 DRL 方法。

In this review, we highlight the most widely explored research directions in DRL for multi-UAV systems and present other research directions that are either neglected or not fully explored. We also emphasize the most significant features associated with the distribution of real multi-UAV applications. Our review is not aimed at providing implementation details or detailed explanations about the multi-agent deep reinforcement learning (MADRL) algorithms but rather is aimed at critically analyzing, through a comparative study, all the common and dissimilar aspects of the various DRL techniques used when addressing multi-UAV problems. Nevertheless, we will provide some basic RL (mainly MARL) concepts as necessary and sufficient conditions to facilitate comprehension and readability.

在本综述中，我们突出了在多无人机系统中广泛探索的 DRL 研究方向，并介绍了其他被忽视或未充分探索的研究方向。我们还强调了与实际多无人机应用分布相关的最显著特征。我们的综述目的不是提供多智能体深度强化学习 (MADRL) 算法的实施细节或详细解释，而是旨在通过比较研究，批判性地分析在解决多无人机问题时使用的各种 DRL 技术的共有和不同之处。尽管如此，我们仍将提供一些基本的强化学习 (主要是多智能体强化学习 MARL) 概念，作为必要和充分条件，以促进理解和可读性。

The rest of this article is structured as follows. In Section 2, we introduce the essential background concepts about single and multi-agent reinforcement learning by highlighting some solutions for cooperation and coordination. Section 3 describes the methodology used to select the works related to DRL-based multi-UAV applications. In Section 4, we classify multi-UAV systems into different classes based on their applications and present a comparison of the most relevant features in each class. In Section 5, a comprehensive discussion about the selected works is presented, whereas Section 6 contains a summary of this review, as well as the conclusions and considerations.

本文其余部分的结构如下。在第 2 节中，我们介绍了关于单一智能体和多智能体强化学习的基本背景概念，通过突出一些用于合作和协调的解决方案。第 3 节描述了用于选择与基于 DRL 的多无人机应用相关工作的方法。在第 4 节中，我们根据应用将多无人机系统分类为不同的类别，并比较了每个类别中最相关的特征。第 5 节对所选工作进行了全面讨论，而第 6 节包含了对本综述的总结，以及结论和思考。

2. Background

2. 背景知识

Some basic and main notions will be provided to understand better how a cooperative multi-UAV system can be investigated when applying deep reinforcement learning techniques. Part of the following definitions is adapted from the book by Sutton and Barto [16] (refer to it for additional insights).

将提供一些基本和主要概念，以更好地理解在应用深度强化学习技术时如何研究协作多无人机系统。以下部分定义参考了 Sutton 和 Barto 的书籍 [16](参见该书以获取更多见解)。

2.1. Single-Agent Markov Decision Process

2.1. 单智能体马尔可夫决策过程

A Markov decision process (MDP) is a mathematical framework modeling sequential decision-making problems. In MDPs, the learner (i.e., the decision maker) is called an agent, and everything that interacts with it is called the environment: a continuous interaction between the agent and the environment takes place during the learning process. An MDP can be formally defined by the tuple $\langle \mathbb{S}, \mathbb{A}, \mathcal{T}, \mathcal{R} \rangle$:

马尔可夫决策过程 (MDP) 是一种数学框架，用于建模顺序决策问题。在 MDP 中，学习者 (即决策者) 被称为智能体，与其交互的一切被称为环境: 在学习过程中，智能体和环境之间进行持续的互动。MDP 可以正式定义为以下元组 $\langle \mathbb{S}, \mathbb{A}, \mathcal{T}, \mathcal{R} \rangle$:

- $\mathbb{S}$ is a set of states $\langle s_0, \ldots, s_n \rangle$, with $s_i \in \mathbb{S}$;

- $\mathbb{S}$ 是一组状态 $\langle s_0, \ldots, s_n \rangle$ ，其中 $s_i \in \mathbb{S}$ ；

- $\mathbb{A}$ is the a set of actions $\langle a_0, \ldots, a_m \rangle$ with $a_i \in \mathbb{A}$ possible in each state s ;

- $\mathbb{A}$ 是一组动作 $\langle a_0, \ldots, a_m \rangle$ ，在每个状态 $a_i \in \mathbb{A}$ 中有 s 种可能的动作；

- $\mathcal{T} : \mathbb{S} \times \mathbb{A} \times \mathbb{S} \rightarrow [0, 1]$ is the probability transition matrix representing the probability of switching from the state s at time t to the state s' at time $t + 1$ by picking the action a ;

- $\mathcal{T} : \mathbb{S} \times \mathbb{A} \times \mathbb{S} \rightarrow [0, 1]$ 是概率转移矩阵，表示在时间 t 从状态 s 转换到时间 $t + 1$ 的状态 s' 的概率，通过选择动作 a 。

- $\mathcal{R} : \mathbb{S} \times \mathbb{A} \times \mathbb{S} \rightarrow \mathbb{R}$ is the reward function that returns the reward obtained by transitioning from state s to s' by picking action a .

- $\mathcal{R} : \mathbb{S} \times \mathbb{A} \times \mathbb{S} \rightarrow \mathbb{R}$ 是奖励函数，它返回通过选择动作 a 从状态 s 转移到 s' 获得的奖励。

where $\mathcal{P}$ and $\mathcal{R}$ represent the environmental dynamics and the long-term rewards, respectively; they also define the world model, i.e., the model of the considered environment. The full state space can always be observed in a single-agent problem (i.e., full observability of the MDP).

其中 $\mathcal{P}$ 和 $\mathcal{R}$ 分别代表环境动态和长期奖励；它们还定义了世界模型，即所考虑环境的模型。在单智能体问题中 (即 MDP 的完全可观测性)，总是可以观察到完整的状态空间。

2.2. Multi-Agent Markov Decision Process

2.2. 多智能体马尔可夫决策过程

A multi-agent Markov decision process is referred to as a Markov game [17], i.e., a generalization of the standard MDP (used in SARL problems) to the multi-agent scenario. A Markov game is defined by a tuple $\langle \mathcal{P}, \mathbb{S}, \mathbb{A}, \mathcal{T}, \mathcal{R} \rangle$:

多智能体马尔可夫决策过程被称为马尔可夫博弈 [17]，即标准 MDP(用于 SARL 问题) 在多智能体场景下的推广。马尔可夫博弈由一个元组 $\langle \mathcal{P}, \mathbb{S}, \mathbb{A}, \mathcal{T}, \mathcal{R} \rangle$ 定义:

- $\mathcal{P}$ is the number of $n \in \mathcal{N}$ players (agents);
- $\mathcal{P}$ 是 $n \in \mathcal{N}$ 玩家 (智能体) 的数量；
- $\mathbb{S}$ is the set of environmental states shared by all the agents $\langle s_0, \ldots, s_k \rangle$ with $s_i \in \mathbb{S}$;
- $\mathbb{S}$ 是所有智能体 $\langle s_0, \ldots, s_k \rangle$ 共享的环境状态集合，其中 $s_i \in \mathbb{S}$ ；
- $\mathbb{A}$ is the set of joint actions $\langle A_1 \times A_2 \cdots \times A_n \rangle$ with $A_i \in \mathbb{A}$, where A_i is the set of actions of agent i ;
- $\mathbb{A}$ 是联合动作集合 $\langle A_1 \times A_2 \cdots \times A_n \rangle$ ，其中 $A_i \in \mathbb{A}$ ，A_i 是智能体 i 的动作集合；
- $\mathcal{T} : \mathbb{S} \times \mathbb{A} \times \mathbb{S} \to [0, 1]$ is the probability transition matrix representing the probability of switching from the state s to s' by picking the joint action $\mathbf{a}$;
- $\mathcal{T} : \mathbb{S} \times \mathbb{A} \times \mathbb{S} \to [0, 1]$ 是概率转移矩阵，表示通过选择联合动作 **a** 从状态 s 转移到 s' 的概率；
- $\mathcal{R}$ is the set of rewards $\langle \mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_N \rangle$, where $\mathcal{R}_i : \mathbb{S} \times \mathbb{A} \times \mathbb{S} \to \mathbb{R}$ is the reward function related to the agent i , which returns the reward obtained by transitioning from the state s to the state s' by taking the action $\mathbf{a}$.
- $\mathcal{R}$ 是奖励集合 $\langle \mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_N \rangle$ ，其中 $\mathcal{R}_i : \mathbb{S} \times \mathbb{A} \times \mathbb{S} \to \mathbb{R}$ 是与智能体 i 相关的奖励函数，它返回通过采取动作 **a** 从状态 s 转移到状态 s' 获得的奖励。

In this case, the MDP can be classified into three main groups: (i) fully observable MDP (FOMDP), where the agent can access the full state space; (ii) partially observable MDP (POMDP), in which the agent only knows local state information; (iii) mixed observability MDP (MOMDP), allowing the agent to obtain both the global and local states.

在这种情况下，MDP 可以分为三大类:(i) 完全可观测 MDP(FOMDP)，其中智能体可以访问完整的状态空间；(ii) 部分可观测 MDP(POMDP)，其中智能体只知道局部状态信息；(iii) 混合可观测 MDP(MOMDP)，允许智能体获取全局和局部状态。

2.3. Single-Agent Reinforcement Learning (SARL)

2.3. 单智能体强化学习 (SARL)

Reinforcement learning (RL) is a machine learning technique where autonomous learners (agents) are supposed to take the best action based on a specific desired goal and by interacting with an environment. In a single-agent reinforcement learning (SARL) system, only one agent acts within the environment by modifying it through its actions. This problem is usually formalized through the single-MDP framework. The learner receives a reward according to the action selected, and its main general goal is choosing actions in such a way as to maximize a cumulative reward over time. One of the main challenges is setting a good trade-off between the exploration phase, which pushes the learner to choose different actions in order to observe new possible paths in the operative environment, and the exploitation phase, which pushes the learner instead to continue to select actions that have been already proven successful. When using RL, the goal of the agent is to maximize a numerical reward signal over time:

强化学习 (RL) 是一种机器学习技术，在这种技术中，自主学习者 (智能体) 需要根据特定的期望目标和与环境的交互来采取最佳行动。在单智能体强化学习 (SARL) 系统中，只有一个智能体在环境中通过其行动对其进行修改。这个问题通常通过单一 MDP 框架形式化。学习者根据所选行动获得奖励，其主要通用目标是选择行动以最大化随时间的累积奖励。主要挑战之一是在探索阶段和利用阶段之间设置一个好的权衡，探索阶段推动学习者选择不同的行动以观察操作环境中的新可能路径，而利用阶段则推动学习者继续选择已经证明成功的行动。使用 RL 时，智能体的目标是最大化随时间的数值奖励信号:

$$G_t = R_{t+1} + R_{t+2} + R_{t+3} + \ldots + R_T$$

where T is the final time step. In order to prevent the previous function from blowing up in the long term and thus focusing on the effect of the short-term actions, the previous formula can be rewritten as follows:

其中 T 是最终时间步。为了防止前面的函数在长期内爆炸，从而关注短期行动的效果，前面的公式可以重写如下:

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \ldots = \sum_{k=0}^{T} \gamma^k R_{t+k+1}$$

where $0 \leq \gamma \leq 1$ is called the discounted rate.

其中 $0 \leq \gamma \leq 1$ 被称为折扣率。

An RL agent uses the previous cumulative reward to improve a policy, a state-value function, or a state-action value function. A policy π represents a mapping function linking a state to an action distribution, and $\pi(a \mid s)$ defines the probability that the agent will pick the action a in the state s .

RL 智能体使用之前的累积奖励来改进策略、状态值函数或状态-动作值函数。策略 π 表示一个将状态映射到动作分布的函数，并且 $\pi(a \mid s)$ 定义了智能体在状态 s 中选择动作 a 的概率。

The value function or state-value function estimates how good it is for an agent to be in a given state. A value function of a state s under the policy π is defined as follows:

价值函数或状态值函数估计了智能体处于给定状态时的优劣。在策略 π 下，状态 s 的价值函数定义为如下:

$$V_\pi(s) = \mathbb{E}_\pi\left[G_t \mid \mathcal{S}_t = s\right]$$

where $\mathbb{E}_\pi$ is the expected value obtained starting from state s and successively choosing actions using the policy π

其中 $\mathbb{E}_\pi$ 是从状态 s 开始并且依次使用策略 π 选择动作所获得的期望值。

The Q-function or action-value function instead estimates how good it is to choose an action a in a given state s . The action-value function of a state s and an action a under the policy π is defined as follows:

Q 函数或动作值函数则是估计在给定状态下选择一个动作 a 的优劣。在策略 π 下，状态 s 和动作 a 的动作值函数定义为如下:

$$Q_\pi(s, a) = \mathbb{E}_\pi\left[G_t \mid \mathcal{S}_t = s, \mathcal{A}_t = a\right]$$

where $\mathbb{E}_\pi$ is the expected value obtained starting by selecting the action a in the given state s and then following the policy π . Reinforcement learning algorithms may be classified in different ways using different criteria, one of which depends on the function type learned by the agent:

其中 $\mathbb{E}_\pi$ 是在给定状态下选择动作 a 并然后遵循策略 π 所获得的期望值。强化学习算法可以根据不同的标准以不同的方式进行分类，其中一种分类依据是智能体学习的函数类型:

- Policy-based methods. The agent learns an explicit representation of a policy π during the learning process;

- 基于策略的方法。智能体在学习过程中学习策略 π 的显式表示；

- Value-based methods. The agent learns a value function to derive an implicit policy π . The selected actions are the ones that maximize the value function;

- 基于价值的方法。智能体学习一个价值函数来推导隐式策略 π 。选择那些使价值函数最大化的动作；

- Actor-critic methods. The agent learns both a policy function and a value function. These methods are mainly related to deep reinforcement learning and can be considered a mix of the previous ones.

- 演员-评价者方法。智能体同时学习策略函数和价值函数。这些方法主要与深度强化学习相关，可以看作是前述方法的混合。

We can further distinguish among the other two main general RL approaches, namely model-based and model-free solutions. In the former, the model of the environment is already given and partially updated over time during the learning process, while in the latter, the environment model is unknown (the agent interacts with it through a trial-and-error procedure by trying to learn its model from scratch).

我们可以进一步区分另外两种主要的通用强化学习 (RL) 方法，即基于模型 (model-based) 和无需模型 (model-free) 的解决方案。在前者中，环境的模型已经给定，并在学习过程中随时间部分更新；而在后者中，环境模型是未知的 (代理通过尝试从头开始学习其模型的试错过程与之交互)。

Finally, we report here, below, the main MDP assumptions associated with most SARL problems:

最后，我们在下面报告了与大多数 SARL 问题相关的主要马尔可夫决策过程 (MDP) 假设:

- Past State Independence. Future states depend only on the current state;

- 过去状态独立性。未来状态仅依赖于当前状态；

- Full Observability. The MDP is often considered to be a FOMDP;

- 完全可观测性。MDP 通常被认为是完全可观测马尔可夫决策过程 (FOMDP)；

- Stationary Environment. $\mathcal{P}$ and $\mathcal{R}$ are constant over time.

- 稳定环境。$\mathcal{P}$ 和 $\mathcal{R}$ 随时间保持不变。

For what concerns deep reinforcement learning (DRL) techniques, we can simply and generically describe them as methods combining the usage of neural networks (NNs) as function approximators with RL frameworks in order to allow the agent to learn more generalizable and scalable behavior.

关于深度强化学习 (DRL) 技术，我们可以简单且泛化地描述为结合使用神经网络 (NNs) 作为函数逼近器与强化学习框架的方法，以允许代理学习更通用和可扩展的行为。

2.4. Multi-Agent Reinforcement Learning (MARL)

2.4. 多代理强化学习 (MARL)

A multi-agent reinforcement learning (MARL) problem can be modeled through a Markov game framework, where multiple agents act simultaneously by shaping the environment accordingly. In MARL problems, as for the SARL case, the agents try to improve a policy according to a specific desired task. However, the policy is obviously associated with a system of agents. Sometimes, it can also happen that each agent improves only its policy selfishly without taking into account the goal of the whole system, either because it is required for a very specific and limited case or because it is competing against the other agents involved in the considered scenario. Based on the type of training and execution paradigm used, which can be centralized or decentralized, three main schemes for multi-agent reinforcement learning can be identified [18]:

多代理强化学习 (MARL) 问题可以通过马尔可夫博弈框架来建模，在该框架中，多个代理通过相应地塑造环境来同时行动。在 MARL 问题中，与 SARL 情况类似，代理试图根据特定期望任务改进策略。然而，策略显然与代理系统相关。有时，也可能发生每个代理仅自私地改进自己的策略，而不考虑整个系统的目标，这可能是因为特定且有限的案例要求，或者是因为它在与考虑的场景中涉及的其它代理竞争。基于使用的训练和执行范式类型，可以是集中式或分布式，可以识别出三种主要的多代理强化学习方案 [18]:

- Centralized training with centralized execution (CTCE) provides a single-cooperative system policy but relies on stable communication among all the agents. The need for a centralized unit makes this approach unmanageable when the number of agents becomes large. Standard SARL algorithms may be adapted to work in this setting;

- 集中训练与集中执行 (CTCE) 提供了单一协作系统策略，但依赖于所有代理之间的稳定通信。由于需要集中单元，当代理数量变得庞大时，这种方法变得难以管理。标准的 SARL 算法可以适应在这种环境下工作；

- Decentralized training with decentralized execution (DTDE) allows independent agents to perform their policies without neither communication nor cooperation. This paradigm is sometimes referred to as independent learning, and SARL algorithms (just as in the previous case) can be used for any individual agent;

- 分散训练与分散执行 (DTDE) 允许独立代理在不进行通信和协作的情况下执行它们的策略。这种范例有时被称为独立学习，SARL 算法 (与前一种情况一样) 可以用于任何单个代理；

- Centralized training with decentralized execution (CTDE), where the agents can access global info during the training time but not at the execution time. The system cooperation is ensured by the centralized training, while the distributed agents execution can be performed without the need for communication, hence providing a more adaptive behavior with respect to the non-stationarity of the environment (see Figure 2 for a schematic visualization);

- 集中训练与分散执行 (CTDE)，其中代理在训练时间内可以访问全局信息，但在执行时间则不可以。系统协作通过集中训练得到保证，而分布式代理的执行可以在无需通信的情况下进行，因此相对于环境的非平稳性，提供了更加自适应的行为 (参见图 2 的示意图)；

A fourth mixed paradigm can also be considered together with the previous ones. It can be represented by a combination of all those approaches, mixing the already mentioned paradigms and/or combining them with other additional techniques explained in Section 2.5.

第四种混合范例也可以与前几种一起考虑。它可以表示为所有这些方法的组合，混合了前面提到的范例和/或将它们与其他在第 2.5 节中解释的附加技术相结合。

Moreover, based on the structure of the reward function, the MARL problem can be defined as follows:

此外，基于奖励函数的结构，多代理强化学习 (MARL) 问题可以定义如下:

- Fully Cooperative. Each agent receives the same reward at each time step $\mathcal{R}_t^1 = \mathcal{R}_t^2 =$

- 完全协作。每个代理在每个时间步接收相同的奖励 $\mathcal{R}_t^1 = \mathcal{R}_t^2 =$

$\ldots = \mathcal{R}_t^n$

- Fully Competitive (or zero-sum). The sum of the rewards of all the agents is zero

- 完全竞争 (或零和)。所有代理的奖励总和为零

$\mathcal{R}_t^1 + \mathcal{R}_t^2 + \ldots + \mathcal{R}_t^n = 0$

- Partially Cooperative. A part of the reward is shared among agents, but they could have additional individual rewards, i.e., $\mathcal{R}^i{}_{\text{Tot}} = \mathcal{R}_{sh} + \mathcal{R}^i{}_{\text{ind}}$, where $\mathcal{R}_{sh}$ is the shared part of the reward and $\mathcal{R}^i{}_{\text{ind}}$ represents the individual reward associated with agent i ;

- 部分协作。部分奖励在代理之间共享，但它们可能有额外的个体奖励，即 $\mathcal{R}^i{}_{\text{Tot}} = \mathcal{R}_{sh} + \mathcal{R}^i{}_{\text{ind}}$ ，其中 $\mathcal{R}_{sh}$ 是奖励的共享部分，$\mathcal{R}^i{}_{\text{ind}}$ 表示与代理 i 相关联的个体奖励；

- Mixed. No specific constraint on the structure of the reward function.

- 混合。对奖励函数的结构没有特定约束。

Finally, we want to highlight the most relevant MARL challenges:

最后，我们希望强调最相关的 MARL 挑战:

- Partial Observability. Action selection takes place under incomplete assumptions;

- 部分可观测性。在存在不完整假设的情况下进行动作选择；

- Scalability. The state and action spaces increase exponentially with the number of agents;

- 可扩展性。状态和行为空间随着代理数量的增加而指数级增长；

- Non-stationarity. All the agents learn and change their policies at the same time;
- 非静态性。所有代理同时学习和改变它们的策略;
- Credit Assignment. Every agent needs to understand its contribution to the joint system reward.
- 信用分配。每个代理都需要理解其对联合系统奖励的贡献。

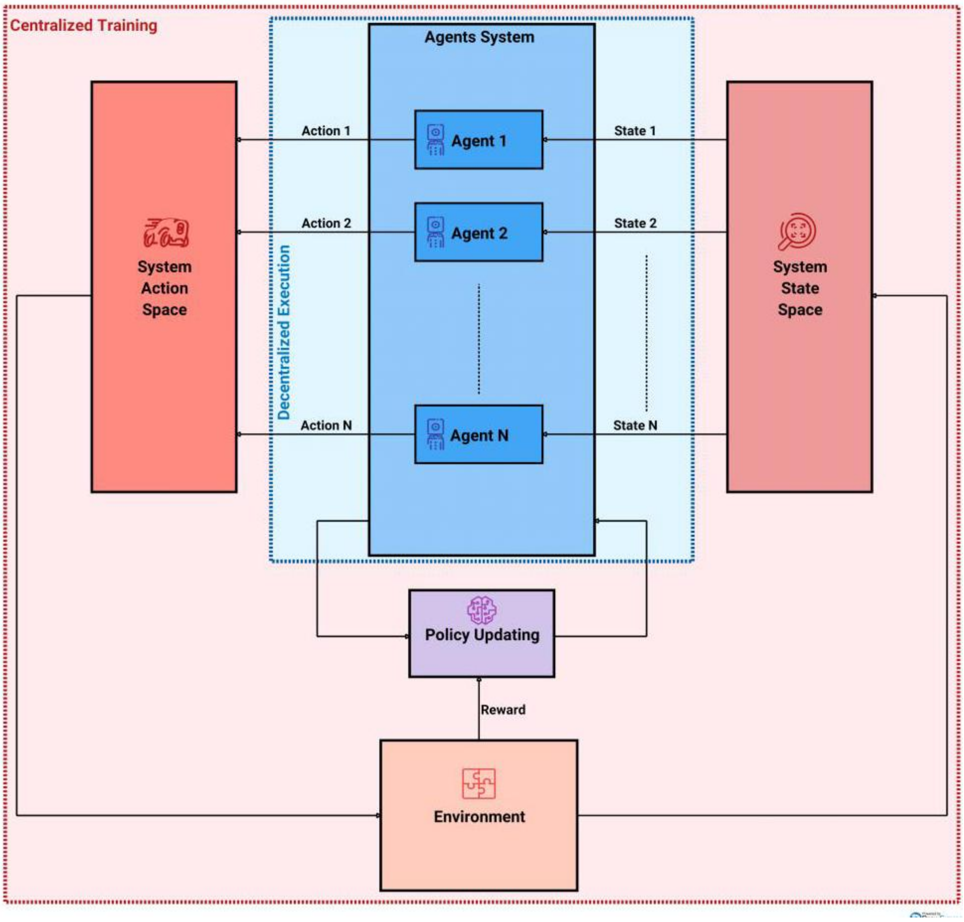


Figure 2. Centralized training decentralized execution (CTDE) paradigm scheme (this image was generated through Draw Express Lite [19]).

图 2. 集中式训练分布式执行 (CTDE) 范式方案 (此图像通过 Draw Express Lite [19] 生成)。

2.5. Additional Techniques

2.5. 额外技术

In this article, the reader may come across specific techniques not necessarily based on (D)RL, such as path planning; mean field; attention mechanism; and, more generally, deep learning strategies. Here, we provide a brief definition of these techniques and additional references that may be useful to clarify their mechanism.

在本文中，读者可能会遇到一些不一定是基于 (D)RL 的特定技术，例如路径规划；均值场；注意力机制；以及更一般意义上的深度学习策略。在这里，我们为这些技术提供简短的定义，以及可能有助于阐明其机制的额外参考文献。

Path planning can be applied globally or locally. The associated techniques allow the agents to find a path from a starting location to a target one by avoiding obstacles (if any). The best possible path can be found based on some specific desired goal, such as energy-saving maximization, traveling time minimization, or even exploration tasks. Some of the most known path planning algorithms are represented by algorithms such as A^*, D^* , RRT, RRT^* , and Dijkstra (see the survey by Karur et al. [20] for an in-depth explanation).

路径规划可以全局或局部应用。相关的技术允许代理找到从起始位置到目标位置的路径，同时避开障碍 (如果有的话)。最佳可能的路径可以根据某些特定的期望目标找到，例如节能最大化、旅行时间最小化，甚至是探索任务。一些最著名的路径规划算法包括 A^*, D^* , RRT, RRT^* 以及 Dijkstra(参见 Karur 等人的调查 [20] 以深入了解)。

Mean field techniques are based on the mean-field game theory. The idea behind them is to decompose a general and complex problem containing many interacting units in a simpler problem where only local interactions between a small number of units are considered. In RL applications, units are usually referred to as agents.

均值场技术基于均值场博弈理论。它们背后的思想是将包含许多相互作用的单元的一般复杂问题分解为一个更简单的问题，其中只考虑少数单元之间的局部相互作用。在 RL 应用中，单元通常被称为代理。

The attention mechanism [21] is used in deep learning techniques to keep the most relevant features of specific input sequences by discharging the less significant ones: this is performed through a weighted combination of all the input vectors.

注意力机制 [21] 在深度学习技术中用于保留特定输入序列的最相关特征，通过释放不太重要的特征来实现: 这是通过对所有输入向量的加权组合来完成的。

Some other techniques can be considered hybrid as they use other deep learning approaches on top of (D)RL methods and/or a combination of the latter.

其他一些技术可以被认为是混合型的，因为它们在 (D)RL 方法之上使用了其他深度学习方法，并且/或者后者的组合。

3. Articles Selection Methodology

3. 文章选择方法

Here, we describe the methodology used to filter all the studies characterized by the features of our interest.

在这里，我们描述了用于筛选所有具有我们感兴趣特征的研究的方法。

Initially, we investigated significant conferences (e.g., International Conference on Autonomous Agents and Multiagent Systems AAMAS, the European Conference on Machine Learning ECML, The International Conference on Machine Learning ICML, and Neural Information Processing Systems NeurIPS). However, since conferences do not allow for an indexed and detailed search, we moved to a keyword search on some of the most relevant scientific databases (e.g., IEEE Xplore and ScienceDirect). The review process started in early 2022, and we initially focused our search on collecting articles from 2013 onwards by combining, through boolean operators, the following keywords: UAV, multi-UAV, reinforcement learning, drone, multi-agent. Different websites offer different advanced search utilities, such as keyword searching in the abstract, in the full text, in metadata, etc. We noticed that searching the keywords in the full text of the document usually led to misleading results, as the topic of some of the papers found was not related to multi-UAV RL-based systems: this happened even though one or more keywords still appeared in the sections associated with the introduction, the reference, or the related works. Searching instead for the presence of the keywords only in the abstract section yielded more significant results.

起初，我们调查了一些重要的会议 (例如，国际自主代理和多代理系统会议 AAMAS，欧洲机器学习会议 ECML，国际机器学习会议 ICML，以及神经信息处理系统会议 NeurIPS)。然而，由于会议不允许进行索引和详细搜索，我们转而在一些最相关的科学数据库 (例如 IEEE Xplore 和 ScienceDirect) 上进行关键词搜索。审查过程始于 2022 年初，我们最初将搜索重点放在通过布尔运算符结合以下关键词收集 2013 年及以后的论文: 无人机 (UAV)，多无人机 (multi-UAV)，强化学习，无人机 (drone)，多代理 (multi-agent)。不同的网站提供了不同的高级搜索工具，例如在摘要、全文、元数据中等进行关键词搜索。我们注意到在文档的全文中搜索关键词通常会导致误导性的结果，因为一些找到的论文的主题与多无人机基于 RL 的系统无关: 即使一个或多个关键词仍然出现在引言、参考文献或相关工作相关的部分，也会出现这种情况。相反，只在摘要部分搜索关键词则产生了更有意义的结果。

With all these things considered, we resolved to search only for the keywords multi-UAV and reinforcement learning in the abstract section of the papers collection related to the year 2022 (the last inspection was on 18 December 2022). Indeed, similar but different keywords such as Drone and UAV often returned articles on single-agent applications, thus not what we were looking for. Due to the extremely high number of works found in the literature during the paper-collection phase, we decided to apply the following additional ”filters” in order to select the papers in which we were interested specifically:

考虑了所有这些因素后，我们决定仅在 2022 年相关论文集的摘要部分搜索关键词“多无人机 (multi-UAV)”和“强化学习 (reinforcement learning)”(最后一次检查是在 2022 年 12 月 18 日)。实际上，类似但不同的关键词，如“无人机 (Drone)”和“UAV”，通常返回的是关于单代理应用的文章，这并非我们所需。由于在文献搜集阶段发现的著作数量极其庞大，我们决定应用以下额外的“筛选器”，以选择我们特别感兴趣的论文:

- Scalable and cooperative approaches. The multi-agent reinforcement learning solution should use either a centralized training decentralized execution (CTDE) scheme (see Section 2.4 for more details) or even a fully decentralized (FD) approach. However, in the latter case, it should be augmented through communication and/or a shared reward(SR)among the agents. When the reward is shared, communication cannot even be present. When the agents are sharing a reward (even if partially), they can learn to collaborate without communication and solve the non-stationarity issue. All the works using either only a centralized training/centralized execution or a decentralized training/decentralized execution paradigm will not be considered: we want our agents to be able to act individually and to cooperate at the same time without being necessarily constrained to the global (and maybe not always accessible) state system information during the execution phase unless a specific communication method is provided. We will also not take into account any works in which the training phase is performed sequentially by keeping fixed the policy of all the other UAVs involved during the learning step and, thus, actually not solving the non-stationary online problem (see Section 2 for more details). Collaboration can be considered as such only if all the UAVs can obtain either the same reward or different rewards but share some common observations through direct (Figure 3a) or indirect (Figure 3b) communication. The observation communicated can result either from the concatenation of all the agents' individual observations (i.e., global communication, referred to as GC) or from a local and/or partial info exchange (i.e., local communication, referred as *LC*): if the shared observations are local, they should not necessarily be partial. Info exchange can be about any feature (e.g., sensor detections, gradients, and q-values). When the training phase is centralized, then cooperation is intrinsically guaranteed. Even if the communication module is not based on a real transmission protocol, this is still a valid feature for UAVs' cooperation. Our criteria reject all the cases where there is no clear info about how the global observations are supposed to be locally available on the agent side at execution time. Even though some works study multi-UAV systems resulting in a final collaborative task, they were not selected if the agents involved are not aware of each other and, thus, every time that the cooperation could be implicitly derived only from the interaction between the environment objects and the agents. For example, the latter case could happen in some specific application scenarios where different users can be served only by one agent at a time; these specific interactions implicitly lead to cooperation among agents, which will then avoid all the already served users. These first filtering choices are meant to be focused on scalable solutions (i.e., without a central control unit at execution time) and cooperative approaches through the usage of shared resources or communications. Indeed, it is well known that completely centralized approaches suffer from the curse of dimensionality;

- 可扩展和协作的方法。多代理强化学习解决方案应使用集中训练分布式执行 (CTDE) 方案 (参见第 2.4 节了解更多细节)，或甚至完全分布式 (FD) 方法。然而，在后者的情况下，它应该通过代理间的通信和/或共享奖励 (SR) 进行增强。当奖励被共享时，通信甚至可以不存在。当代理共享奖励 (即使是部分的)，它们可以在没有通信的情况下学习协作并解决非平稳性问题。所有仅使用集中训练/集中执行或分布式训练/分布式执行范式的作品将不被考虑: 我们希望我们的代理能够独立行动并同时协作，而不必在执行阶段受到全局 (且可能并不总是可访问) 的状态系统信息的约束，除非提供了特定的通信方法。我们也不会考虑那些在训练阶段通过保持所有其他参与学习的无人机的策略固定来按顺序执行的作品，因此实际上并没有解决非平稳的在线问题 (参见第 2 节了解更多细节)。只有当所有无人机可以获得相同的奖励或不同的奖励但通过直接 (图 3a) 或间接 (图 3b) 通信共享一些共同观察结果时，协作才可以被认为是这样的。所通信的观察结果可以是所有代理个体观察结果的串联 (即全局通信，简称 GC)，也可以是局部和/或部分信息交换 (即局部通信，简称 *LC*): 如果共享观察结果是局部的，它们不一定是部分的。信息交换可以是任何特征 (例如，传感器检测、梯度、q 值)。当训练阶段是集中时，合作本质上得到保证。即使通信模块不是基于真实的传输协议，这仍然是无人机合作的有效特征。我们的标准排除了所有关于全局观察如何在执行时在代理端局部可用的信息不明确的情况。尽管有些作品研究导致最终协作任务的多无人机系统，但如果涉及的代理彼此不知情，那么这些作品将不被选中，因此每次合作只能从环境对象与代理之间的交互隐含导出。例如，在某些特定的应用场景中，不同的用户可能只能由一个代理一次服务；这些特定的交互隐含地导致代理之间的合作，从而避免已经服务的所有用户。这些初步筛选选择旨在关注可扩展解决方案 (即在执行时没有中央控制单元) 和通过使用共享资源或通信的协作方法。事实上，

众所周知，完全集中式方法受到维度诅咒的影响；

- Deep RL. We only consider reinforcement learning approaches that use function approximators based on deep neural networks. This choice is guided by the fact that we want to focus on generalizable solutions: in real environments, some unpredictable events which are not experienced during the learning phase can occur, and RL techniques cannot handle them as smoothly as DRL approaches can;

- 深度强化学习 (Deep RL)。我们只考虑基于深度神经网络函数逼近器的强化学习方法。这一选择是基于我们希望关注泛化性解决方案的事实: 在真实环境中，可能会发生学习阶段未经历过的一些不可预测事件，而强化学习技术无法像深度强化学习 (DRL) 方法那样平滑地处理它们；

- Number of UAVs. The number of UAVs effectively used in the experiments should be at least greater than two (comparison analyses including a number of drones ranging from 2 to N , with N > 2 , have still been taken into account). For the sake of clarity, we highlight that we filtered an article out even though it was focused on a system (which could be heterogeneous) made up of a number of agents equal to or greater than three (but with a number of UAVs involved in the considered system not greater than two). In addition, we did not select any papers where the agents are not represented by the UAVs even if UAVs are somehow involved (indeed, sometimes, it could happen that the agents are represented by a central controller such as a server or a base station);

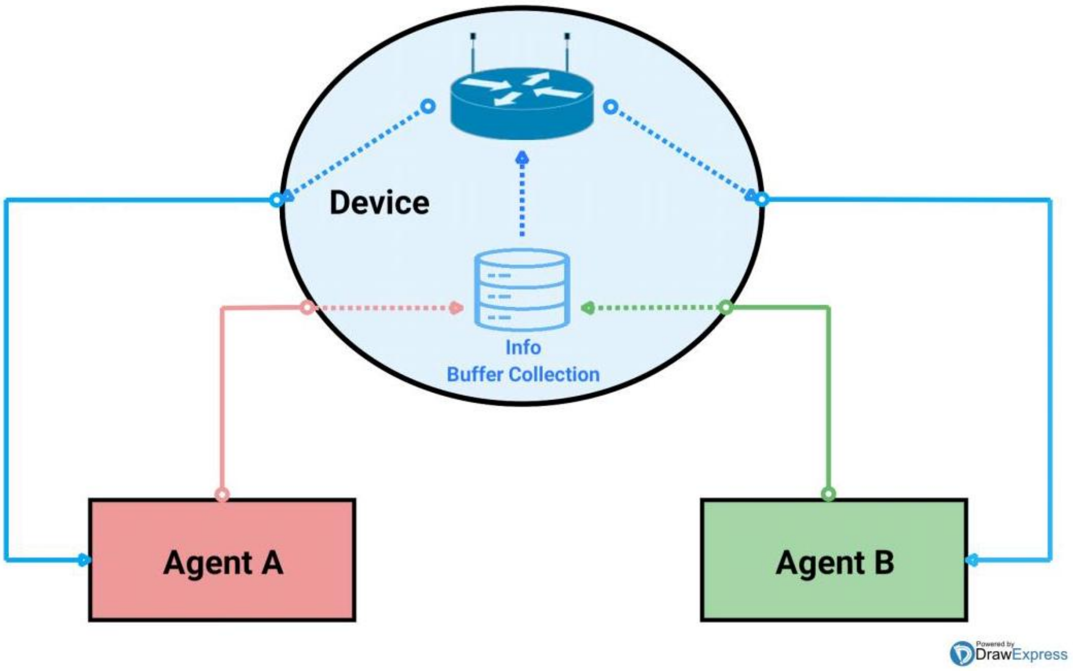
- 无人机的数量 (Number of UAVs)。实验中实际使用的无人机数量应至少大于两架 (已经考虑了从 2 到 N 架无人机，包括 N > 2 的比较分析)。为了清晰起见，我们强调，尽管一篇文章关注的是一个由至少三个代理组成的系统 (可能是异构的)，但我们还是将其过滤掉了 (但考虑到系统中的无人机数量不超过两架)。此外，我们没有选择任何论文，其中代理不是由无人机表示，即使无人机以某种方式参与其中 (确实，有时可能发生的是，代理由中央控制器如服务器或基站表示)；

- Clear description of the framework. The MDP and the framework structure must be explicitly defined. If not, their main features (i.e., state space, action space, or reward function) must be clearly or easily derivable from the context or figures.

- 框架的清晰描述。MDP 和框架结构必须明确定义。如果没有，它们的主要特征 (即状态空间、动作空间或奖励函数) 必须从上下文或图中清晰地或容易推导出来。

Agent A
Agent B
Powered by DrawExpress

Figure 3. Cont.
图 3。续。



(b) Agent to Device (A2D).

Figure 3. A2A (a) and A2D (b) communications: (a) represent a direct communication between agents, while in (b), communication takes place through a device, which can be a server, a central communication node (supervisor), or any device that is supposed to collect info from each agent and then broadcast the global state to all the agents. Only two agents have been represented here for easy representation, but there can be more than two. These images were generated through Draw Express Lite [19].

图 3. A2A(a) 与 A2D(b) 通信:(a) 表示代理之间的直接通信，而在 (b) 中，通信通过一个设备进行，该设备可以是服务器、中心通信节点 (监督者) 或任何旨在从每个代理收集信息然后向所有代理广播全局状态的设备。这里仅为了易于表示而展示了两个代理，但实际上可能超过两个。这些图像是通过 Draw Express Lite [19] 生成的。

4. DRL-Based Multi-UAV Applications

4. 基于深度强化学习的多无人机应用

In this section, we show the main features of the selected works, which have been divided into five main classes, assigned according to each multi-UAV application considered:

在本节中，我们展示了所选作品的主要特点，这些作品被分为五个主要类别，根据每个考虑的多无人机应用进行分配:

- Coverage;
- 覆盖；
- Adversarial Search and Game;
- 对抗搜索与游戏；
- Computational Offloading;
- 计算卸载；
- Communication;
- 通信；

- Target-Driven Navigation.

- 目标驱动导航。

Before moving on to the multi-UAV classes analysis, we want to highlight some particular aspects that can help better understand what follows. First of all, the terms UAVs, drones, and agents are used in an interchangeable way and with the same meaning.

在进行多无人机类别分析之前，我们想强调一些特定方面，以帮助更好地理解接下来的内容。首先，无人机 (UAVs)、无人机 (drones) 和代理 (agents) 这些术语在本文中可以互换使用，并且具有相同的含义。

Every class feature comparison table (e.g., Table 1) shows different features associated with each application class. The following features will be considered in every multi-UAV class (some useful abbreviations that could be used for these features are reported in the abbreviation list in Section 6):

每个类别特征比较表 (例如，表 1) 显示了与每个应用类别相关的不同特征。在每一个多无人机类别中都将考虑以下特征 (在第六节的缩写列表中报告了这些特征可能使用的一些有用缩写):

- Algorithm indicates the algorithm used by the authors;

- 算法，指作者使用的算法；

- Paradigm refers to the learning paradigm used:

- 范式指的是使用的学习范式:

- Centralized training with decentralized execution (CTDE);

- 集中训练与分布式执行 (CTDE)；

- Fully decentralized (FD) with either local communication (FDLC), global communication (FDGC), or shared reward (FDSR). A combination of some or all the previous ones is also possible (e.g., FDLCSR or FDGCSR);

- 完全分布式 (FD)，可以是局部通信 (FDLC)、全局通信 (FDGC) 或共享奖励 (FDSR)。也可以是前面某些或所有方法的组合 (例如，FDLCSR 或 FDGCSR)；

- No. of UAVs points out the number of UAVs considered in the experiments;

- 无人机的数量指出实验中考虑的无人机数量；

- Collision avoidance is specified by a flag value (either a check mark or a cross) indicating whether collision avoidance is taken into account;

- 碰撞避免由一个标志值 (勾选或叉号) 指定，表示是否考虑碰撞避免；

- Action space is associated with a categorical value, i.e., it can be specified by discrete (D), continuous (C), or hybrid (H) in order to indicate the value type of the considered action space. The type of action space is not indicated whenever the authors do not clearly define it or not smoothly inferrable (a cross will be present in this particular case);

- 动作空间与分类值相关，即它可以由离散 (D)、连续 (C) 或混合 (H) 指定，以表示考虑的动作空间的值类型。当作者没有明确定义或不能平滑推断动作空间类型时，动作空间类型不予指示 (在这种情况下将出现叉号)；

- State space: as for the action space, but here, it is referred to as the state space;

- 状态空间: 与动作空间类似，但这里指的是状态空间；

- $3D$ is a flag value indicating if the UAVs can move in a three-dimensional space or not (i.e., only 2D);

- $3D$ 是一个标志值，表示无人机是否能在三维空间中移动 (即仅限 2D)；

- UAV dynamic model shows whether a UAV dynamic model is used (a check mark will be present) or not (a cross will be reported even if the UAV dynamic model is either not specified or not clearly inferable). The dynamic model must include low-level control inputs such as forces and/or torques and, thus, take into account also the mass of the UAVs. All the works using a simulator provided with realistic UAVs models, e.g., AirSim [22] and Gazebo [23], will be considered as solutions that include the UAV dynamic model by default. This feature does not apply to any studies using the dynamic model of a material point, even when including the forces and/or torques applied on the UAVs' center of mass;

- UAV 动态模型显示了是否使用了 UAV 动态模型 (将出现勾选标记) 或者没有使用 (即使 UAV 动态模型未指定或无法明确推断，也将报告交叉标记)。动态模型必须包括低级别控制输入，如力与/或扭矩，因此，还需考虑 UAV 的质量。所有使用配备有现实 UAV 模型的模拟器的研究，例如 AirSim [22] 和 Gazebo [23]，将被视为默认包含 UAV 动态模型的解决方案。此功能不适用于使用物质点动态模型的研究，即使它们包括作用在 UAV 质心上的力与/或扭矩；

- Propulsion energy consumption is specified by a flag value (either a check mark or a cross), indicating whether the energy consumption due to the flight phase is considered or not.

- 推进能量消耗由一个标志值 (勾选标记或交叉标记) 指定，表示是否考虑了飞行阶段的能量消耗。

Other task-dependent features vary from class to class. They will be indicated by * in the associated class table (their meaning will be explained inside each dedicated class section).

其他与任务相关的特征因类别而异。它们将在相关类别表中用 * 标示 (它们的意义将在每个专用类别部分内解释)。

It is worth mentioning that the number of UAVs (indicated by No. of UAVs) shown inside the comparative tables follows the rules described below:

值得一提的是，比较表中显示的无人机数量 (由“无人机的数量”表示) 遵循以下规则:

- $i-j$ points out that a number of UAVs between i and j (it could also include all the integer numbers among them) is used for the considered case;

- $i-j$ 指出在考虑的案例中使用了 i 到 j 之间的无人机数量 (其中可能包括它们之间的所有整数)；

- $i, j, k, \dots$ indicates instead that a number of UAVs respectively and exactly equal to $i, j, k, \dots$ is tested for the specific considered case.

- $i, j, k, \dots$ 则表示在特定考虑的案例中分别且确切地测试了等于 $i, j, k, \dots$ 的无人机数量。

Finally, we considered as a 2D case every scenario in which the UAVs were assumed to fly at a fixed and constant height. Indeed, even though the UAVs' height is specified, if it is fixed, the movement along the Z -axis is not actually considered.

最后，我们将假定无人机在固定和恒定高度飞行的情况视为 2D 案例。实际上，尽管指定了无人机的飞行高度，但如果它是固定的，那么沿 Z 轴的运动并未真正考虑。

4.1. Coverage

4.1. 覆盖率

In this section, we consider all those applications involving a fair distribution of UAVs with respect to a series of requirements.

在本节中，我们考虑所有涉及公平分配无人机 (UAVs) 以满足一系列要求的那些应用。

The coverage task is usually associated with a fairness index that requires all points of interest (PoI) to be treated fairly without prioritizing some over others. The models of the environments in the coverage category share a common structure. More in detail, a number $\mathcal{N} = \{n = 1, 2, \dots, N\}$ of UAVs with a limited sensing range and a set $\mathcal{K} = \{k = 1, 2, \dots, K\}$ of PoIs to be covered is considered (usually $K > N$); additional constraints may include obstacle avoidance and limited battery. One of the main functions used to bring geographical fairness is Jain's Fairness index:

覆盖任务通常与一个公平性指数相关联，该指数要求所有兴趣点 (PoI) 公平对待，不优先考虑某些点。覆盖类别中的环境模型具有共同的结构。更具体地说，考虑了一定数量 $\mathcal{N} = \{n = 1, 2, \dots, N\}$ 的具有有

限感知范围的无人机和一组 $\mathcal{K} = \{k = 1, 2, \ldots, K\}$ 需要被覆盖的兴趣点 (通常 $K > N$)；附加约束可能包括避障和电池有限。用于实现地理公平性的一个主要函数是 Jain 的公平性指数:

$$\left[f_t = \frac{\left(\sum\limits_{k=1}^{K} c_t\left(k\right) \right)^2}{K \sum\limits_{k=1}^{K} c_t(k)^2} \right] \tag{1}$$

where $c_t\left(k\right)$ is a generic function that evaluates the coverage score of a specific PoI k at time step t . 其中 $c_t\left(k\right)$ 是一个通用函数，用于评估特定兴趣点 k 在时间步 t 的覆盖得分。

The target applications for this class are usually related to mobile crowd sensing (MCS) [24,25] and mobile-assisted edge computing (MEC) [26]. In MCS, a group of UAVs equipped with sensors must collect data in contexts such as smart cities, emergency rescues, agricultural productions, solar panel inspections, and many others. In MEC applications, a team of UAVs support ground user equipment in order to offload part of their tasks. Such a scenario can be modeled with different levels of complexity by considering or not additional constraints such as collision avoidance, energy consumption, dynamic PoI, or unconstrained UAVs' movement on a 3D plane.

这类目标应用通常与移动群感知 (MCS)[24,25] 和移动辅助边缘计算 (MEC)[26] 相关。在 MCS 中，一组配备传感器的无人机必须在智能城市、紧急救援、农业生产、太阳能板检测等多种环境中收集数据。在 MEC 应用中，一群无人机支持地面用户设备，以分担其部分任务。这种场景可以通过考虑或不考虑附加约束 (如避碰、能耗、动态 PoI 或无人机在三维平面上的无约束运动) 来以不同复杂度建模。

Table 1 contains different features associated with the coverage class. In particular, the PoI Model represents works in which the points of interest are not static and predefined at the beginning, but their dynamics are explicitly modeled.

表 1 包含与覆盖类别相关的不同特征。特别是，PoI 模型表示那些兴趣点在开始时不是静态和预定义的，而是明确建模其动力学的工作。

Table 1. Coverage class features. The mark $*$ indicates class-dependent features and is thus considered only for this class.

表 1。覆盖类别特征。标记 $*$ 表示类别依赖特征，因此仅适用于此类。

Ref.	Algorithm	Par.	No. of UAVs	Col. Av.	Prop. Eg.	Action Space	State Space	3D	UAV Dyn.	PoI Model *
[24]	EDICS	CTDE	2–5	✓	✓	C	H	X	X	X
[27]	DDPG	FDGCSR	3–8	X	✓	C	C	X	X	X
[26]	MADDPG	CTDE	3,4	✓	✓	C	H	X	X	X
[28]	PPO	FDSR	3,4	X	3	D	C	X	X	✓
[29]	HGAT	FDLC	10,20,30,40	X	✓	D	C	X	X	X
[30]	SBG-AC	FDSR	3–9	✓	✓	D	C	✓	X	X
[25]	DRL-eFresh	CTDE	up to 50	✓	✓	C	H	X	X	X
[31]	MFTRPO	FDLC	5–10	X	✓	C	H	X	X	X
[32]	SDQN	CTDE	10	X	X	D	D	X	X	X

参考文献	算法	段落	无人机数量	列表平均	推断示例	行动空间	状态空间	3D	无人机动力学	兴趣点模型 *
[24]	工程与科学信息分类字典	连续时间动态环境	2–5	✓	✓	C	H	X	X	X
[27]	深度确定性策略梯度	稳定化模糊动态高斯过程回归	3–8	X	✓	C	C	X	X	X
[26]	多智能体深度确定性策略梯度	连续时间动态环境	3,4	✓	✓	C	H	X	X	X
[28]	近端策略优化	稳定化动态高斯过程回归	3,4	X	3	D	C	X	X	✓
[29]	HGAT(图注意力网络的一种变体)	FDLC(深度学习中的循环神经网络)	10,20,30,40	X	✓	D	C	X	X	X
[30]	SBG-AC(基于策略梯度的深度确定性策略梯度算法)	稳定化动态高斯过程回归	3–9(可能表示范围或序号)	✓	✓	D	C	✓	X	X
[25]	DRL-eFresh(深度强化学习在新鲜度优化中的应用)	连续时间动态环境	up to 50(高达 50)	✓	✓	C	H	X	X	X
[31]	MFTRPO(多任务信任区域策略优化算法)	FDLC(深度学习中的循环神经网络)	5–10(可能表示范围或序号)	X	✓	C	H	X	X	X
[32]	SDQN(双样本差分 Q 网络)	连续时间动态环境	10	X	X	D	D	X	X	X

In this category, the work of Nemer et al. [30] is the only one to explicitly consider the 3D movement of UAVs in the DRL algorithm. The authors proposed a distributed solution to address the challenge of fair coverage and energy consumption for UAVs to provide communication to ground users. In their scenario, UAVs are free to move in a 3D space, but the ground region is divided into cells, and the center of each cell is a PoI. Each UAV is required to cover a number of PoIs for a certain time. They modeled both the channels for communication and energy consumption. The problem is formulated as a state-based potential game, and it is combined with the DDPG algorithm; the authors refer to this solution as State-Based Game with Actor-Critic (SBG-AC). This approach was able to compete and sometimes outperform both the centralized DRL-EC3 [33] and distributed Distributed-DRL [27] under different metrics such as the average coverage score, the fairness index, and the normalized average energy consumption. Instead, Bai et al. [28] are the only ones considering dynamic PoIs. Here, the goal is to use a team of UAVs to provide fair communication quality of service (QoS) to all the users, whose movement is defined through a random walk model. Although the collision avoidance task is not taken into account, obstacles are present and may obstruct communication. They used the PPO algorithm, training a single network with the experience of all the UAVs, and then, they distributed the network to all the UAVs during the testing phase. Thus, the authors define their method as CTDE, but we argue that this approach can be defined as a decentralized approach with parameter sharing and cooperative reward. In addition to the geographical fairness, in Wang et al. [26], the authors also take into consideration the fairness of load balance at each UAV. More in detail, multi-UAV-assisted mobile edge computing is here considered, where UAVs support ground user equipment (UEs) by allowing them to offload part of their tasks. The UEs (and not the UAVs) decide to offload or to compute a task locally based on a minimum energy consumption principle: this is why we did not include this work in the computational offloading class. The goal is to maximize the geographical fairness among the UEs, the fairness of UE-load at each UAV, and to minimize the energy consumption for all the UEs. Similarly to the previous case, a CTDE solution for the mobile crowd sensing (MCS) problem is provided by Liu et al. [24]. Here, both UAVs and driverless cars are used as mobile terminals (MTs) to collect data. They are constrained by their carrying capacity and sensing range: the main objective is to provide an efficient route for the MTs given a sensing area. The authors proposed an energy-efficient distributed mobile crowd sensing (Edics) algorithm based on MADDPG with an N-step return and a prioritized experience replay. They compared Edics with four baseline algorithms (including the classical MADDPG approach) and used metrics such as energy efficiency, geographical fairness, data collection, and energy consumption ratios. They were able to outperform the baseline in almost every scenario.

在此类别中，Nemer 等人 [30] 的工作是唯一明确考虑了无人机 (UAV) 在深度强化学习 (DRL) 算法中 3D 移动的研究。作者提出了一种分布式解决方案，以解决无人机公平覆盖和能耗的挑战，为地面用户提供通信。在他们的场景中，无人机可以在 3D 空间中自由移动，但地面区域被划分为单元格，每个单元格的中心是一个兴趣点 (PoI)。要求每架无人机在特定时间内覆盖一定数量的 PoI。他们建立了通信通道和能耗的模型。问题被表述为一个基于状态的潜在博弈，并与深度确定性策略梯度 (DDPG) 算法结合；作者将这个解决方案称为基于状态的演员-评论家游戏 (SBG-AC)。这种方法能够在不同的指标下与集中式的 DRL-EC3 [33] 和分布式的分布式 DRL [27] 竞争，有时甚至超越它们，例如平均覆盖得分、公平指数和归一化平均能耗。而 Bai 等人 [28] 是唯一考虑动态 PoI 的研究者。在这里，目标是使用一组无人机为所有用户提供公平的通信服务质量 (QoS)，用户的移动通过随机游走模型定义。尽管没有考虑避障任务，但存在障碍物可能会阻碍通信。他们使用了 PPO 算法，训练一个包含所有无人机经验的单一网络，然后，在测试阶段将网络分发给所有无人机。因此，作者定义他们的方法为 CTDE，但我们认为这种方法可以定义为具有参数共享和合作奖励的分布式方法。除了地理公平性，Wang 等人 [26] 的研究中还考虑了每个无人机负载平衡的公平性。更具体地说，这里考虑了多无人机辅助移动边缘计算，其中无人机通过允许地面用户设备 (UEs) 卸载部分任务来支持它们。UEs(而不是无人机) 根据最小能耗原则决定是卸载任务还是本地计算: 这就是为什么我们没有将这项工作归类为计算卸载类别。目标是最大化 UEs 之间的地理公平性、每个无人机上 UE 负载的公平性，以及最小化所有 UEs 的能耗。与之前的情况类似，Liu 等人 [24] 为移动 crowd sensing(MCS) 问题提供了一个 CTDE 解决方案。在这里，无人机和无驾驶汽车都被用作移动终端 (MTs) 来收集数据。他们的携带能力和感知范围受限: 主要目标是给定一个感知区域，为 MTs 提供一个有效的路线。作者提出了一种基于 MADDPG 的节能分布式移动 crowd sensing(Edics) 算法，具有 N 步回报和优先级体验回放。他们将 Edics 与四个基线算法 (包括经典的 MADDPG 方法) 进行了比较，并使用了诸如能效、地理公平性、数据收集和能耗比率等指标。他们几乎在所有场景中都能超越基线。

Similarly, an MCS scenario is also considered by Dai et al. [25], but differently from the previous case, the authors also add a constraint on data freshness by setting a duration deadline for the validity of the sensed information. In particular, the UAVs need to maximize the amount of data collected and the geographical fairness and to minimize energy consumption by avoiding obstacles and guaranteeing the freshness of data. The starting point for their proposed solution is the proximal policy optimization algorithm (PPO), called DRL-eFresh: a shared global PPO model is trained while multiple workers receive a copy of the global PPO model in order to interact independently with the environment. A different approach for the coverage problem based on game theory is given by Chen et al. [31]. The main goal here is to provide fair coverage to a set of ground users (or PoIs) while minimizing the UAVs' energy consumption. To solve this problem, the authors formalized it as a mean field game (MFG) and used the Hamilton-Jacobi-Bellman/Fokker-Planck-Kolmogorov (HJB/FBP) equation to find the Nash-mean field equilibrium (Nash-MFE). Nevertheless, this approach presents some problems as it is required to solve stochastic partial differential equations (SPDEs), and the MFE is obtained through two coupled SPDEs whose formulation is prone to human error. Since the authors noticed that the solution to the HJB/FPK equation could also be obtained by using neural networks and reinforcement learning, they proposed a solution based on a combination of MFG and trust region policy optimization (TRPO), named as Mean-Field TRPO (MFTRPO). An agent collects the state of the environment using the action from MFTRPO and sends information to its neighbors.

同样，Dai 等人 [25] 也考虑了 MCS 场景，但与之前的情况不同，作者们还增加了一个关于数据新鲜度的约束，为感知信息的有效性设置了一个持续时间截止。特别是，无人机需要最大化收集的数据量、地理公平性，并通过避开障碍物来最小化能耗，保证数据的实时性。他们提出解决方案的起点是邻近策略优化算法 (PPO)，称为 DRL-eFresh: 训练一个共享的全局 PPO 模型，同时多个工作者接收到全局 PPO 模型的副本，以便独立地与环境互动。Chen 等人 [31] 基于博弈论提出了另一种解决覆盖问题的方法。这里的主要目标是向一组地面用户 (或兴趣点) 提供公平的覆盖，同时最小化无人机的能耗。为了解决这个问题，作者们将其形式化为一个均值场游戏 (MFG)，并使用 Hamilton-Jacobi-Bellman/Fokker-Planck-Kolmogorov(HJB/FBP) 方程来找到 Nash 均值场均衡 (Nash-MFE)。然而，这种方法存在一些问题，因为它需要解决随机偏微分方程 (SPDEs)，而 MFE 是通过两个耦合的 SPDEs 获得的，这些方程的公式容易出错。由于作者们注意到 HJB/FPK 方程的解也可以通过使用神经网络和强化学习获得，他们提出了一个基于 MFG 和信任域策略优化 (TRPO) 相结合的解决方案，命名为 Mean-Field TRPO(MFTRPO)。一个代理使用 MFTRPO 的动作收集环境的状态，并向其邻居发送信息。

Liu et al. [27] consider UAVs as mobile base stations (BSs) to provide long-term communication coverage for ground mobile users. The authors use an actor-critic approach with a centralized critic. The target networks are not just copied from the main networks but are slowly updated. The UAVs receive a common reward based on average coverage fairness, which helps them to work cooperatively. With this approach, the authors obtained a better performance than their previous state-of-the-art approach $DRL - EC^3$ [33], which was based on a fully centralized solution. While the previously described solutions were built for ad hoc use cases, Chen et al. [29] provide a generic solution by applying it to two different multi-UAVs operations, i.e., UAV recon and predator-prey scenarios. Below, we will discuss only the design choices related to the UAV recon scenario. Here, the states of the environment are represented as graphs, and two UAVs can communicate only if they are inside a specific communication range. In particular, an agent builds two graphs: an observation graph containing all the entities inside the observation range of the agent and a communication graph containing all the agents inside the communication range of the considered agent. An adjacency matrix is then built for both of these graphs. An encoder layer gives the network structure for both entities and agents inside a group, followed by two hierarchical graph attention (HGAT) modules, one for the entities inside the observation range and one for the entities inside the communication range. The output of the two HGATs is then concatenated and passed on to a recurrent unit. They used an actor-critic framework and conducted experiments with up to 40 UAVs, showing the scalability of their approach.

刘等人 [27] 将无人机视为移动基站 (BSs)，为地面移动用户提供长期通信覆盖。作者使用了一种带有集中式评判者的演员-评判者方法。目标网络不仅仅是简单地从主网络复制，而是缓慢更新。无人机根据平均覆盖公平性接收到的通用奖励，有助于它们进行协作。采用这种方法，作者比他们之前基于完全集中式解决方案的领先方法 $DRL - EC^3$ [33] 获得了更好的性能。而之前描述的解决方案是为特定临时用例设计的，陈等人 [29] 通过将其应用于两种不同的多无人机操作，即无人机侦察和捕食者-猎物场景，提供了一个通用解决方案。下面，我们将仅讨论与无人机侦察场景相关的设选择。在这里，环境的状态表示为图，只有当两个无人机在特定的通信范围内时，它们才能进行通信。特别是，一个智能体构建了两个图: 一个包含智能体观察范围内所有实体的观察图，以及一个包含考虑智能体通信范围内所有智能体的通信图。然后为这两个图构建邻接矩阵。编码器层为组内的实体和智能体提供网络结构，之后是两个分层图注意力 (HGAT) 模块，一个用于观察范围内的实体，另一个用于通信范围内的实体。两个 HGAT 的输出随后被连接并传递给循环单元。他们使用了一个演员-评判者框架，并进行了最多 40 架无人机的实验，展示了他们方法的可扩展性。

A more complex scenario is presented by Mou et al. [32], where the goal is to cover a 3D irregular terrain surface through a two-level UAV hierarchy made up of high-level leaders UAVs (LUAVs) and low-level follower UAVs (FUAVs). Each sub-swarm of UAVs uses a star network communication topology where the leader acts as the communication center and can communicate with all the FUAVs. In contrast, the FUAVs can only communicate with their LUAVs. LUAVs have their leader communication network (LCN), where two LUAVs can establish a communication link to communicate with each other if their signal power exceeds a certain threshold. The reinforcement learning agents are represented by the LUAVs, which are supposed to select the next position in the discretized environment. Although the problem is described as a 3D coverage scenario, the reinforcement learning algorithm is used to learn a policy to select one of four possible directions: therefore, we considered it a 2D model in Table 1. The reward comprises a shared part assessing the overall coverage rate of the entire terrain and connectivity conditions of the LCN and a private part punishing the LUAV for revisiting the positions covered previously. The DRL algorithm proposed here, called Swarm Deep Q-learning (SDQN), is based on the value decomposition method and, thus, structured through a CTDE paradigm. Beyond the specific reinforcement learning algorithm used, as we have already seen in previous papers, some other techniques were adopted, such as parameter sharing [28], prioritized experience replay [24,26], N-step return [24], and the usage of target networks [24,26,27,30].

Mou 等人 [32] 提出了一个更复杂的场景，目标是利用由高层领导无人机 (LUAVs) 和低层跟随无人机 (FUAVs) 组成的两级无人机层次结构来覆盖 3D 不规则地形表面。每个无人机子群使用星型网络通信拓扑，其中领导者作为通信中心，可以与所有 FUAVs 通信。相比之下，FUAVs 只能与它们的 LUAVs 通信。LUAVs 拥有它们的领导者通信网络 (LCN)，如果两个 LUAVs 的信号功率超过某个阈值，它们可以建立通信链路以相互通信。强化学习代理由 LUAVs 表示，它们需要在离散环境中选择下一个位置。尽管问题被描述为一个 3D 覆盖场景，但强化学习算法用于学习选择四个可能方向之一的策略: 因此，我们在表 1 中将其视为 2D 模型。奖励包括一个评估整个地形覆盖率和 LCN 连通条件的共享部分，以及一个惩罚 LUAV 重新访问之前已覆盖位置的私人部分。这里提出的 DRL 算法，称为群体深度 Q 学习 (SDQN)，基于价值分解方法，因此，通过 CTDE 范式构建。除了我们已经在前文中看到的特定强化学习算法外，还采用了其他一些技术，例如参数共享 [28]、优先经验回放 [24,26]、N 步回报 [24] 以及目标网络的使用 [24,26,27,30]。

Eventually, it is worth noticing that different architectures are considered in addition to the standard fully connected layers. In particular, convolutional neural networks (CNNs) [24,25] are used to extract valuable information when data are represented as matrices, while recurrent networks are integrated to take historical data into account [25,29]. Hierarchical graph attention layers (HGAT) are also applied in order to combine both graph neural networks and the attention mechanism [29].

最终，值得注意的是，除了标准的全连接层之外，还考虑了不同的架构。特别是，卷积神经网络 (CNNs)[24,25] 用于在数据表示为矩阵时提取有价值的信息，而循环网络被整合以考虑历史数据 [25,29]。为了结合图神经网络和注意力机制，还应用了分层图注意力层 (HGAT)[29]。

4.2. Adversarial Search and Game

4.2. 对抗搜索与游戏

This class includes all the scenarios where two opposing UAV teams compete against each other. Even though from an outer perspective there is a winning and a losing UAV team and hence a zero-sum rewards structure, internal cooperation among the agents of the same team is required to defeat the opponents. Relevant features belonging to this class are referred to as attack model and learning enemies in Table 2. When two teams compete against each other, it could happen that only one team (the ”good” team) is learning a specific behavior, while the other team (referred to as the ”enemy” team) is not learning any policy. Thus, learning enemies indicates whether the opponent team is somehow learning and improving a policy over time either by using another algorithm or even by using the same algorithm as that used by the ”good” team. The attack model is related instead to the explicit design of a model to decree whether an attack has been successful.

这一类包括所有两个对立的无人机队伍相互竞争的场景。尽管从外部视角来看，有一个胜利的无人机队伍和一个失败的无人机队伍，因此存在零和奖励结构，但同一队伍中的代理之间的内部合作是击败对手所必需的。表 2 中提到的与这一类相关的特征被称为攻击模型和学习敌人。当两个队伍相互竞争时，可能发生的情况是只有一个队伍 (被称为“好”的队伍) 在学习特定的行为，而另一个队伍 (被称为“敌”的队伍) 没有学习任何策略。因此，学习敌人表示对方队伍是否在某种程度上通过使用另一种算法或甚至使用与“好”队伍相同的算法，随时间学习和改进策略。攻击模型则与设计一个模型来判定攻击是否成功的明确设计有关。

Table 2. Adversarial search and game scenario class features. The mark * indicates class-dependent features and is thus considered only for this class: Att. Mod. and Learn. Enem. stand, respectively, for attack model and learning enemies.

表 2. 对抗搜索与游戏场景类特征。标记 * 表示依赖类别的特征，因此仅在此类别中考虑:Att. Mod. 和 Learn. Enem. 分别代表攻击模型和学习敌人。

Ref.	Algorithm	Par.	No. of UAVs	Act. Space	St. Space	3D	UAV Dyn.	Col. Av.	Prop. Eg	Att. Mod.*	Learn. Enem.*
[34]	MADDPG	CTDE	4	D	D	✓	X	X	X	✓	X
[35]	MADDPG	CTDE	2,3	D	X	✓	X	X	X	✓	✓
[36]	WMFQ+ WMFAC	FDLC	25,50,100	D	D	X	X	X	X	✓	✓
[29]	HGAT	FDLC	10,20,30,4	D	D	X	X	X	X	X	✓
[37]	MADDPG	CTDE	3	C	C	X	X	X	X	✓	✓

Ref.(参考文献)	Algorithm(算法)	段落	无人机的数量	行动空间	状态空间	3D	无人机动力学	集群可用性	推理示例	注意力模型 *	学习对手模型 *
[34]	多智能体深度确定性策略梯度 (MADDPG)	连续时间确定性策略估计 (CTDE)	4	D	D	✓	X	X	X	✓	X
[35]	多智能体深度确定性策略梯度 (MADDPG)	连续时间确定性策略估计 (CTDE)	2,3	D	X	✓	X	X	X	✓	✓
[36]	WMFQ+ WMFAC	FDLC	25,50,100	D	D	X	X	X	X	✓	✓
[29]	HGAT	FDLC	10,20,30,4	D	D	X	X	X	X	X	✓
[37]	多智能体深度确定性策略梯度 (MADDPG)	连续时间确定性策略估计 (CTDE)	3	C	C	X	X	X	X	✓	✓

Wang et al. [36] tackled the problem of UAV swarm confrontation. Using decentralized approaches based only on local observation leads to the problem of non-stationarity since multiple agents are learning simultaneously. In this work, the situation information of a UAV is usually determined by itself and its neighboring UAVs. In particular, the authors applied the mean field theory to model the communication between UAVs and considered a virtual UAV as an average of its neighboring agents' value functions to model the interaction between the allies (the number of neighbor agents is predefined). Additionally, different weight coefficients were assigned to different neighboring UAVs based on an attention mechanism. This approach allows for handling the dynamism in the number of the UAVs due to the fact that some of them are removed from the scenario during the match: an enemy UAV has a certain probability of being destroyed if it is inside the attacking range of the opponent UAV, which corresponds to a circular sector. Two algorithms using mechanism attention are proposed here, i.e., the weighted Mean Field Q-learning (WMFQ) and the Weighted Mean Field Actor-Critic (WMFAC) approaches. Experiments with a 50vs . 50 UAV swarm confrontation were performed, and the proposed solution was compared with other algorithms such as independent double Q-learning, independent actor-critic, MFQ, and MFAC. Similarly to the previous case, Li et al. [34] defined a cone-shaped attack area in front of the UAV (associated with a certain probability of being hit) based on the distance and the relative position between the UAV and its target. The proposed solution is based on a MADDPG algorithm with a multi-actor-attention-critic structure. The main idea is to learn a centralized critic using an attention mechanism to let the agents learn to whom to pay attention. This approach is referred to as Multi-Agent Transformed-based Actor-Critic (MATAC), and it is based on Bidirectional Encoder Representations from Transformers (BERT) [38]. A gated recurrent unit in the actor network is used in order to use historical information in the decision-making process. Additionally, a dual-experience replay mechanism is used to ease the problem of sparse reward. The authors designed a 4 vs. 4 adversarial environment and compared their solution with multiple state-of-the-art algorithms such as MAAC, COMA, and MADDPG. A different approach based on an incomplete information dynamic game was proposed by Ren et al. [35]. The dynamic game was considered incomplete since the UAVs have missing information, such as the opponent's strategy. In the scenario considered here, the flight trajectories of the UAVs are decomposed into seven elementary maneuvers: the goal is to cooperatively decide which maneuvers best fit to achieve victory against the opponent team. The utility function of the dynamic game, which is given by the pay-off of certain mixed strategies, is also used as the reward for the underlying reinforcement learning algorithm. The maneuver decision is based on a situation assessment model, computed through a situation function and based on spatial geometric relationships between UAVs. The target enemy of a UAV is the one with the largest value related to the situation function. An attack is considered successful if the enemy is within the attack range of the UAV for a certain amount of time. A dynamic Bayesian network was designed to infer the opponent's tactical decision. Just like in the work by Wang et al. [36], the swarm confrontation problem is also tackled by Zhang et al. [37]. Each UAV is associated with an attacking distance and two different angles: the front angle, which represents the UAV attacking zone, and the rear angle, which indicates the UAV unprotected zone. A UAV can defeat the opponent if the former is within the attacking angle and distance of the considered UAV, which must be inside the unprotected zone of the opponent. Initially, the authors used the standard MADDPG algorithm and, then, added two additional techniques. The first is the scenario-transfer learning, which uses experience from one scenario to another. In particular, a two-step transfer learning was used with one of the teams not learning while the other one learns and vice versa. The second additional approach is represented by the usage of a self-play technique, which allows the agents to improve by playing against themselves. The teams were trained one at a time. The proposed model was tested in a 3 vs. 3 scenario by showing that the suggested solution has a faster convergence with respect to the standard MADDPG algorithm without using either scenario transfer or self-play techniques. The last work in this section is by Chen et al. [29], whose recon scenario part has already been described in the coverage class. We will now cover the part of the paper associated with the predator-prey scenario, where there is a certain number of predator and prey UAVs, with the predator UAVs having a lower speed than the prey UAVs. The prey is considered caught if its distance from a predator is below a certain threshold, while it is considered safe if a certain amount of time without being caught has elapsed. The action space is the same as the one in the coverage scenario, while the reward depends on the time needed to catch all the prey UAVs. The predator and prey UAVs were trained using the previously described HGAT algorithm and other DRL techniques (e.g., DQN and MADDP).

王等人 [36] 解决了无人机群对抗的问题。仅基于局部观察的分布式方法导致了非平稳性问题，因为多个代理同时在学习。在这项工作中，无人机的情况信息通常由其本身及其邻近的无人机确定。特别是，作者应用了均值场理论来建模无人机之间的通信，并将虚拟无人机视为其邻近代理价值函数的平均值，以建模盟友之间的互动 (邻居代理的数量是预定义的)。此外，根据注意力机制，为不同的邻近无人机分配了

不同的权重系数。这种方法允许处理由于一些无人机在比赛过程中被移除导致的无人机数量的动态变化: 如果敌方无人机在对手无人机的攻击范围内，它有一定的概率被摧毁，这对应于一个圆形扇区。本文提出了两种使用注意力机制的算法，即加权均值场 Q 学习 (WMFQ) 和加权均值场演员-评论家 (WMFAC) 方法。进行了 50 个无人机群对抗的实验，并将所提出的解决方案与其他算法进行了比较，如独立的双重 Q 学习、独立演员-评论家、MFQ 和 MFAC。与之前的情况类似，李等人 [34] 根据无人机与其目标之间的距离和相对位置，在无人机前方定义了一个锥形攻击区域 (与被击中的概率相关)。所提出的解决方案基于具有多演员-注意力-评论家结构的 MADDPG 算法。主要思想是使用注意力机制学习集中式评论家，让代理学会关注谁。这种方法被称为基于变换的多代理演员-评论家 (MATAC)，并且基于双向编码器表示来自变换器 (BERT)[38]。演员网络中的门控循环单元用于在决策过程中使用历史信息。此外，使用双重经验回放机制来缓解稀疏奖励问题。作者设计了一个 4 对 4 的对立环境，并将他们的解决方案与多种最先进的算法进行了比较，如 MAAC、COMA 和 MADDPG。Ren 等人 [35] 提出了一个基于不完整信息动态游戏的不同方法。动态游戏被认为是非完整的，因为无人机缺失了一些信息，例如对手的策略。在考虑的情景中，无人机的飞行轨迹被分解为七个基本机动: 目标是合作决定哪些机动最适合对抗对手团队。动态游戏的效用函数，由特定混合策略的收益给出，也被用作底层强化学习算法的奖励。机动决策基于情况评估模型，该模型通过无人机之间的空间几何关系计算得出。无人机的主要目标是与情况函数相关值最大的敌方无人机。如果敌人在无人机的攻击范围内停留一定时间，则认为攻击成功。设计了一个动态贝叶斯网络来推断对手的战术决策。与王等人 [36] 的工作类似，张等人 [37] 也解决了无人机群对抗问题。每架无人机都与一个攻击距离和两个不同的角度相关联: 前方角度代表无人机的攻击区，后方角度表示无人机的无保护区。如果前者在考虑的无人机的攻击角度和距离内，并且必须位于对手的无保护区内，无人机可以击败对手。最初，作者使用了标准的 MADDPG 算法，然后增加了两种额外的技术。第一种是场景迁移学习，它将一个场景的经验用于另一个场景。特别是，使用了两步迁移学习，其中一个团队不学习，而另一个团队学习，反之亦然。第二种额外方法是使用自我博弈技术，它允许代理通过自我对抗来提高。团队是逐个训练的。提出的模型在一个 3 对 3 的场景中进行了测试，结果表明，建议的解决方案在没有使用场景迁移或自我博弈技术的情况下，比标准 MADDPG 算法收敛更快。本节的最后一项工作是陈等人 [29] 的，其中关于侦察场景的部分已经在覆盖类别中描述过。现在我们将介绍与捕食者-猎物场景相关的论文部分，其中有数量一定的捕食者和猎物无人机，捕食者无人机的速度低于猎物无人机。如果猎物无人机与捕食者的距离低于某一阈值，则认为猎物被捕获，如果在一定时间内未被捕获，则认为猎物安全。动作空间与覆盖场景中的相同，而奖励取决于捕获所有猎物无人机所需的时间。捕食者和猎物无人机使用之前描述的 HGAT 算法和其他深度强化学习技术 (例如 DQN 和 MADDP) 进行训练。

4.3. Computation Offloading

4.3. 计算卸载

In modern applications, there is a growing need for remote devices to exchange data with a central node, e.g., a server, in such a way that it can process the information received from any smart device in a significantly reduced time thanks to its much higher computational power. This process is mostly known as computation offloading. In real-time applications that employ autonomous vehicles, a critical aspect is finding a proper trade-off between increasing the vehicles' onboard resources and offloading vehicles' tasks to a cloud computing server. The former could increase the vehicles' manufacturing costs and energy consumption. At the same time, the latter could sometimes be unfeasible due to some possible constraints associated with the wireless communication delay: hence, a compromise is needed. The works described in this section can also handle some system constraints which can be associated with other classes (e.g., coverage and/or communication), but our intention here is to classify in computational offloading all papers where the offloading of some tasks is explicitly mentioned (this is the core goal even if other sub-tasks could be needed).

在现代应用中，远程设备与中心节点 (例如服务器) 交换数据的需求日益增长，这种方式可以利用中心节点远高于智能设备的计算能力，在显著减少的时间内处理从任何智能设备接收到的信息。这一过程通常被称为计算卸载。在实时应用中使用自动驾驶车辆时，一个关键方面是找到增加车辆车载资源与将车辆任务卸载到云计算服务器之间的适当平衡。前者可能会增加车辆的制造成本和能耗。同时，后者有时可能不可行，因为可能与无线通信延迟相关的某些约束: 因此，需要妥协。本节描述的工作也可以处理与其他类别 (例如覆盖和/或通信) 相关的某些系统约束，但我们的意图是将所有明确提到某些任务卸载的论文归类为计算卸载 (即使可能需要其他子任务，这也是核心目标)。

Table 3 shows some features strictly associated with the computation offloading class, which needs to be explained more in detail. The offloading energy (OFLD Eg) is associated with any possible energy consumption related to any offloading task, while OFLD refers to the implementation of a likely computational offloading model. Trajs. indicates instead if the agents' trajectories are either learned (L) or set by default (D); sometimes, it is assumed that UAVs are not moving, and this particular case is referred to as fixed (F).

表 3 展示了一些与计算卸载类别严格相关的特征，这些特征需要更详细地解释。卸载能耗 (OFLD Eg) 与任何与卸载任务相关的能耗有关，而 OFLD 指的是一个可能的计算卸载模型的实现。Trajs. 表示代理的轨迹是学习得到的 (L) 还是默认设置的 (D)；有时，假设无人机不会移动，这种特殊情况被称为固定 (F)。

Table 3. Computation offloading class features. The mark ∗ indicates class-dependent features and is thus considered only for this class. Com. Eg stands for communication energy, while Trajs. means trajectories and OFLD is the abbreviation used for offloading.

表 3. 计算卸载类别特征。标记 ∗ 表示类别依赖特征，因此仅适用于此类。Com. Eg 代表通信能耗，而 Trajs. 表示轨迹，OFLD 是用于卸载的缩写。

Ref.	Algorithm	Par.	No. of UAVs	Col. Av.	Act. Space	State Space	3D	UAV Dyn.	Prop. Eg	OFLD **Eg***	∗ OFLD	∗ Trajs.
[39]	MATD3	CTDE	2,3	✓	C	C	✓	X	X	✓	✓	L
[40]	DON	FDLC	1–10	X	C	C	X	X	X	X	X	F
[41]	MASAC	CTDE	3–6	X	C	C	X	X	✓	✓	✓	D
[42]	MARL AC	FDGC	50,100,150,200	X	D	C	X	X	X	✓	✓	F
[43]	MADDPG	CTDE	2,3,9	✓	C	H	✓	X	✓	X	✓	L
	MADDPG	CTDE	1–6	✓	C	H	✓	X	✓	X	✓	L

参考文献	算法	段落	无人机数量	列表平均值	行动空间	状态空间	3D	无人机动力学	Prop. 示例	OFLD **Eg***	∗ OFLD	∗ Trajs.
[39]	MATD3	CTDE	2,3	✓	C	C	✓	X	X	✓	✓	L
[40]	DON	FDLC	1–10	X	C	C	X	X	X	X	X	F
[41]	MASAC	CTDE	3–6	X	C	C	X	X	✓	✓	✓	D
[42]	MARL AC	FDGC	50,100,150,200	X	D	C	X	X	X	✓	✓	F
[43]	MADDPG	CTDE	2,3,9	✓	C	H	✓	X	✓	X	✓	L
	MADDPG	CTDE	1–6	✓	C	H	✓	X	✓	X	✓	L

Four studies use a CTDE paradigm [39,41,43,44], and only two works [40,41] seem to vary in an appreciable way from some existing algorithms; in the latter, the mobile edge computing (MEC) technology was investigated, and both the centralized and distributed paradigms were proposed. For what concerns the distributed approach used, the authors could obtain agents' cooperation by using Deep Q-Networks (DQNs) with a Q-value transfer mechanism: the i -th UAV can learn a collaborative behavior as its loss function receives the optimal Q-value of neighboring agents at the latest time step. The distributed method here results in being competitive against the centralized one, even if it is worth mentioning that the performance gap in favor of the centralized one increases as the number of UAVs increases. A particular approach was applied by Cheng et al. [41], using a MADDPG framework. However, each UAV was provided with a Soft Actor-Critic (SAC) network instead of the DDPG algorithm usually used in a MADDPG approach: this method, referred to as Multi-Agent Soft Actor-Critic (MASAC), outperforms other well-known algorithms (e.g., SAC, DDPG, and MADDPG) used as baselines in a scenario in which the agents jointly offload tasks and energy to fog computing nodes (FCNs) placed on the ground. It is also worth mentioning the usage of MARL and blockchain technology by Seid et al. [44], where the problem of secure computational offloading with energy harvesting in the context of IoT devices was faced. As UAVs have limited energy resources, blockchain technology was used to secure transactions between resource providers (RPs) and resource requesters (RRs) and to perform operations such as consensus or transaction management. A MADDPG-based algorithm was used, and the proposed solution was evaluated against multiple baselines through different metrics. In computational offloading applications, UAVs are sometimes assumed to not be moving, and the learned actions only include offloading and/or resource allocation strategies, just as the ones used by Liu et al. and Sacco et al. [40,42]. In the latter, the agents were considered as nodes of a graph, where each UAV can communicate with each other through a consensus updating method, and the number of UAVs involved was noticeably larger than the other works belonging to the same class. Additionally, neighbor agents share a local value function. Note also that propulsion and offloading energy consumption were never both taken into account in the same work except for the one proposed by Cheng et al. [41]. Finally, we want to highlight that since the work by Gao et al. (August 2021) [45] does not show significant changes with respect to Gao et al. (December 2021) [43] except for some features (e.g., DDPG instead of MADDPG) and baseline algorithms used, we will consider only the study by Gao et al. (December 2021) [43] in the comparative table (Table 3).

四项研究使用了 CTDE 范式 [39,41,43,44]，只有两篇作品 [40,41] 与现有算法相比似乎有显著的不同；在后者的研究中，探讨了移动边缘计算 (MEC) 技术，并提出了集中式和分布式两种范式。关于所使用的分布式方法，作者通过使用带有 Q 值传递机制的深度 Q 网络 (DQNs) 可以获得代理之间的协作: 第 i 个无人机可以通过其损失函数接收邻近代理在最新时间步的最优 Q 值，学习到协作行为。这里的分布式方法证明其性能可以与集中式方法相媲美，尽管值得注意的是，随着无人机数量的增加，集中式方法的优势性能差距也在增加。Cheng 等人 [41] 应用了一种特别的方法，使用了 MADDPG 框架。然而，每个无人

机都配备了 Soft Actor-Critic(SAC) 网络，而不是 MADDPG 方法中通常使用的 DDPG 算法: 这种方法，称为多代理 Soft Actor-Critic(MASAC)，在代理共同将任务和能量卸载到地面上的雾计算节点 (FCNs) 的场景中，比其他已知算法 (例如，SAC、DDPG 和 MADDPG) 性能更优。还值得提及的是 Seid 等人 [44] 使用了 MARL 和区块链技术，在该研究中，解决了物联网设备背景下带有能量采集的安全计算卸载问题。由于无人机能源资源有限，区块链技术被用来保障资源提供者 (RPs) 和资源请求者 (RRs) 之间的交易安全，并执行诸如共识或交易管理等操作。使用了基于 MADDPG 的算法，并通过不同的指标对所提出的解决方案与多个基线进行了评估。在计算卸载应用中，有时假设无人机不移动，学习到的行为仅包括卸载和/或资源分配策略，正如 Liu 等人及 Sacco 等人 [40,42] 使用的那样。在后者中，将代理视为图的节点，每个无人机可以通过一种共识更新方法与其他无人机通信，涉及的无人机数量明显大于同一类别中的其他研究。此外，相邻代理共享局部价值函数。请注意，除了 Cheng 等人 [41] 提出的研究之外，其他研究均未同时考虑推进和卸载的能量消耗。最后，我们要强调的是，由于 Gao 等人 (2021 年 8 月)[45] 与 Gao 等人 (2021 年 12 月)[43] 相比，除了某些特性 (例如，使用 DDPG 而不是 MADDPG) 和基线算法外，没有显示显著变化，因此我们将在比较表 (表 3) 中仅考虑 Gao 等人 (2021 年 12 月)[43] 的研究。

4.4. Communication

4.4. 通信

Many multi-UAV systems were studied and designed to ease and support communication services. This task usually examines constraints such as energy, computational, and coverage limitations. We include in this class all the works considering the enhancement of a multi-UAV communication-based system regardless of other possible constraints which could be secondary (even if contributing to the achievement of the final goal). This policy is mainly motivated by the fact that some works belonging to the communication class can have intersection features with other classes such as computation offloading and/or coverage. Thus, we consider here only papers for which the main goal (explicitly or implicitly declared) was trying to optimize the performance of a UAV-based communication system through, for example, optimal resource allocation, throughput maximization, Age of Information (AoI) minimization, optimal data harvesting, and/or trajectories optimization.

许多多元无人机系统被研究和设计来简化和支持通信服务。这项任务通常会检查诸如能源、计算和覆盖范围限制等约束条件。我们将所有考虑提升多无人机通信系统性能的工作纳入此类，不考虑其他可能的次要约束 (即使它们有助于最终目标的实现)。这一政策主要是基于这样一个事实: 属于通信类别的一些工作可能与其他类别 (如计算卸载和/或覆盖) 具有交叉特征。因此，我们在这里只考虑那些主要目标 (明确或隐含声明) 是尝试通过例如最优资源分配、吞吐量最大化、信息时龄 (AoI) 最小化、最优数据收集和/或轨迹优化来优化基于无人机的通信系统性能的论文。

Table 4 shows the main features associated with the communication class. The most class-dependent features need additional clarification and are provided below. In particular, we highlight that the communication energy feature is associated with all possible communication (e.g., among UAVs, among users) energy consumption, while $U2U$ and $U2D$ communication models refer to the implementation of a likely communication model for UAV-to-UAV (i.e., direct communication among agents) and UAV-to-device communications (a device can be a represented by anything other than a UAV, e.g., a user or a BS). Trajectories are also considered since the communication class is mainly focused on network aspects. Indeed, sometimes, UAVs' trajectories are assumed to be either predefined, indicated as default (D), or fixed (F), namely motionless UAVs, rather than learned (L).

表 4 显示了与通信类别相关的主要特征。最依赖类别的特征需要额外的解释，如下所述。特别是，我们强调通信能源特征与所有可能的通信 (例如，无人机之间、用户之间) 能源消耗相关，而 $U2U$ 和 $U2D$ 通信模型指的是无人机之间 (即代理之间的直接通信) 和无人机与设备通信 (设备可以是除无人机之外的任何东西，例如用户或基站) 的可能通信模型的实施。轨迹也被考虑在内，因为通信类别主要关注网络方面。实际上，有时无人机的轨迹被假定为预定义的，标记为默认 (D)，或固定 (F)，即静止的无人机，而不是学习得到的 (L)。

Table 4. Communication class features. The mark $*$ indicates class-dependent features and is thus considered only for this class. Com. Eg stands for communication energy, while Trajs. is the abbreviation used for trajectories. $U2U$ and $U2D$ show, respectively, whether a UAV-to-UAV and/or a UAV-to-device communication model is used.

表 4. 通信类别特征。标记 $*$ 表示类别相关特征，因此仅针对此类进行考虑。Com. Eg 代表通信能量，而 Trajs. 是用于轨迹的缩写。$U2U$ 和 $U2D$ 分别显示是否使用了无人机间 (UAV-to-UAV) 和/或无人机到设备 (UAV-to-device) 的通信模型。

Ref. Algorithm	Par.	No. of UAVs	Col. Av.	Act. Sp.	St. Sp.	3D	UAV Dyn.	Prop. Eg	Com. Eg *	∗ U2U	∗ U2D	∗ Trajs.
[46] MAUC	CTDE	2–6	✓	C	C	X	X	✓	X	X	✓	L
[47] DHDRL	FDLC	10,15	X	H	X	X	X	X	X	✓	✓	D
[48] MAHDRL	CTDE	3–7	X	H	H	X	X	X	✓	X	✓	D
[49] DE-MADDPG	CTDE	8,10,12	X	C	C	✓	X	X	X	✓	✓	L
1501 HDRL	CTDE	2–20	✓	D	H	X	X	✓	X	X	✓	L
[51] cDQN	FDGC	3	X	D	D	✓	X	X	X	✓	✓	L
[52] MADDPG	FDGC	1,2,3	✓	C	D	✓	X	X	✓	X	✓	L
[53] DQN	FDGC	10	X	D	D	X	X	X	X	X	✓	L
[54] CA2C	FDGCSR	2–6	X	H	H	X	X	X	X	X	✓	L
[55] MADRL-SA	FDLCSR	2–10	X	H	C	X	X	X	X	X	✓	D
[56] DDPG	FDGC	6,8,10,12	X	C	C	X	X	X	X	X	✓	L
[57] DTPC	CTDE	4	✓	C	C	X	X	✓	✓	X	✓	L
[58] MADRL-based	CTDE	3	✓	H	C	X	X	X	X	X	✓	L
[59] UAFC	CTDE	2,3	X	C	C	X	X	✓	✓	X	√	L
[60] CAA-MADDPG	CTDE	3	X	C	H	✓	X	X	X	X	✓	L

参考文献算法	段落	无人机数量	列平均	动作空间	状态空间	3D	无人机动力学	Prop. 示例	Com. 示例 *	∗ U2U	∗ U2D	∗ Trajs.
[46] MAUC	CTDE	2–6	✓	C	C	X	X	✓	X	X	✓	L
[47] DHDRL	FDLC	10,15	X	H	X	X	X	X	X	✓	✓	D
[48] MAHDRL	CTDE	3–7	X	H	H	X	X	X	✓	X	✓	D
[49] DE-MADDPG	CTDE	8,10,12	X	C	C	✓	X	X	X	✓	✓	L
1501 HDRL	CTDE	2–20	✓	D	H	X	X	✓	X	X	✓	L
[51] cDQN	FDGC	3	X	D	D	✓	X	X	X	✓	✓	L
[52] MADDPG	FDGC	1,2,3	✓	C	D	✓	X	X	✓	X	✓	L
[53] DQN	FDGC	10	X	D	D	X	X	X	X	X	✓	L
[54] CA2C	FDGCSR	2–6	X	H	H	X	X	X	X	X	✓	L
[55] MADRL-SA	FDLCSR	2–10	X	H	C	X	X	X	X	X	✓	D
[56] DDPG	FDGC	6,8,10,12	X	C	C	X	X	X	X	X	✓	L
[57] DTPC	CTDE	4	✓	C	C	X	X	✓	✓	X	✓	L
[58] 基于 MADRL 的	CTDE	3	✓	H	C	X	X	X	X	X	✓	L
[59] UAFC	CTDE	2,3	X	C	C	X	X	✓	✓	X	√	L
[60] CAA-MADDPG	CTDE	3	X	C	H	✓	X	X	X	X	✓	L

Both the CTDE and the decentralized paradigms with communication are quite equally used among the papers in this class. In some works [46,48,52,53,55,56,58], already existing algorithms are directly used, or no significant variations are applied to them: e.g., Qin et al. [46] apply a general MADRL-based distributed UAV-BS Control (MAUC) approach, but actually, it is mainly MADDPG-based as it also uses a centralized critic and decentralized actors (MAUC considers the communication fairness, while other used approaches seem not to take it into account). Even though Zhao et al. [52] use a MADDPG algorithm with experience replay, we can consider it mostly as a classical MADDPG algorithm: similar reasoning can be applied for the work by Wu et al. [53], where a DQN algorithm with experience replay is applied. Another analogous situation is the one faced by Emami et al. [55], where the algorithm used is called Multi-UAV Deep Reinforcement Learning-based Scheduling Algorithm (MADRL-SA), but it is mainly associated with a DQN approach based on agents cooperation; in this latter case, a data gathering problem is faced, and different ground sensors are used to record and share info about other UAVs with the current UAV that is scheduling the sensor (local communication).

在这一类文章中，CTDE 和带有通信的分布式范例的使用相当均衡。在一些工作中 [46,48,52,53,55,56,58]，直接使用了已经存在的算法，或者没有对它们应用显著的变化: 例如，Qin 等人 [46] 应用了一种基于 MADRL 的分布式无人机基站控制 (MAUC) 方法，但实际上，它主要是基于 MADDPG 的，因为它也使用了一个集中式评判者和分布式执行者 (MAUC 考虑了通信公平性，而其他使用的方法似乎没有考虑到这一点)。尽管 Zhao 等人 [52] 使用了带有经验回放的 MADDPG 算法，我们仍然可以认为它主要是一种经典的 MADDPG 算法: 类似的推理也适用于 Wu 等人 [53] 的工作，其中应用了带有经验回放的 DQN 算法。另一种类似的情况是 Emami 等人 [55] 面临的情况，所使用的算法称为基于多无人机深度强化学习的调度算法 (MADRL-SA)，但它主要与基于代理合作的 DQN 方法相关；在后一种情况下，面临的是数据收集问题，不同的地面传感器被用来记录并与其他无人机共享关于当前调度传感器的无人机的信息 (局部通信)。

More appreciable modifications to well-known algorithms can be found instead in other studies [47-51,54,57,59,60] which are analyzed in more depth below. Xu et al. [47] proposed a decomposed heterogeneous DRL (DHDRL) approach to enhance the collaborative backhaul rate (BR) with limited information exchange in UAV networks: a backhaul network is part of a network that generally links a central network to other distributed networks of the same hierarchical network. The used backhaul framework decomposes the joint action space into three submodules (action spaces), which are represented by the transmission target selection (TS), the power control (PC), and the power allocation (PA). More in detail, the TS problem is faced through deep Q-learning (DQL) to decrease the BR (also reducing a useless relay), while the PC and PA problems are solved by a DDPG approach, which allows for matching the traffic requirements with suitable transmission power. Even if each UAV executes each of these three modules with a distributed behavior, it is worth mentioning that U2U communication is implemented, and thus, neighbor UAVs can communicate and cooperate accordingly after receiving a local observation from one-hop neighbor agents: using graph theory, we can define one-hop neighbors as the nodes (i.e., the agents) close to a specific path between a source and a destination node. Cheng et al. [48] referred to their technique as MAHDRL, showing a similar structure to the QMIX algorithm but using a hybrid action space and an actor-critic structure in order to handle it: the joint UAV-user association, channel allocation, and transmission power control (J-UACAPC) problem is faced here. The main idea behind using the MAHDRL approach is to identify the hybrid action pair (i.e., discrete-continuous action pair) generating the maximum action value. A decomposed MADDPG (DE-MADDPG) algorithm was developed by Zhu et al. [49], who studied a scenario involving UAV clusters by rebuilding a connected network where a virtual navigator was proposed to address the instability issue of the state space: the virtual UAV is associated with the barycenter of the cluster of UAVs and thus allows for the relative coordinates to be considered rather than absolute ones. De-MADDPG combines the MADDPG with a single-agent DDPG using reward decoupling, i.e., a global and a local reward. UAVs are represented here as graph nodes, whose edges indicate the communication links between two agents and depend on the distance between each agent. Zhou et al. [50] modeled their problem as a POMPD by using a hybrid DRL approach with a QoE-driven adaptive deployment strategy: multiple UAVs are deployed at specific positions in a target area to serve moving ground users (GUs) through a wireless connection. Each agent is provided with a value network (distributed control) defined as a deep recurrent Q-network (DRQN), while a hybrid network (which is a feedforward NN) is used to pass the output coming from each agent to a layer aimed at taking the final step of the decision-making process (centralized training). In order to speed up the hybrid DRL training phase, the initial UAV deployment is performed through a genetic algorithm (GA), i.e., an algorithm based on a heuristic search inspired by evolution theory (see the review by Katoch et al. [61] for more details); cooperation is achieved by ensuring connectivity and collision avoidance between each UAV. The same scenario with slightly different goals (e.g., coverage maximization instead of QoE maximization) was already dealt with by Ma et al. [62] and further extended by Wu et al. [63], where an attention mechanism is also used. Considering the last three cited papers, only one of them [50] will be inserted into the comparative Table 4 as it provides the most detailed and clear info about the learning structure used, while the other ones result in being simply either a less exhaustive work [62] or a simple extension with not relevant changes except for the introduction of the attention mechanism when using the HDRL technique [63]. The study by Zhang et al. [51] represents instead a particular case in which some coverage constraints are also taken into account, but they are not sufficient (as explained at the beginning of this section) to allow us to classify it into the coverage class. The authors thus proposed a decentralized paradigm (indeed, UAVs act as independent agents) with PS and a constrained DQN (cDQN) algorithm in order to design UAVs trajectories that allow for satisfying some coverage constraints just to improve the capacity of a communication system where ground terminals (GTs) are supposed to be covered by the agents.

在其他研究 [47-51,54,57,59,60] 中，可以找到对已知算法的更值得注意的修改，这些修改在下面进行了更深入的分析。Xu 等人 [47] 提出了一种分解的异质 DRL(DHDRL) 方法，以增强在无人机网络中具有有限信息交换的协作回程速率 (BR): 回程网络通常是连接中心网络和同一层次网络的其他分布式网络的网络的一部分。所使用的回程框架将联合动作空间分解为三个子模块 (动作空间)，分别由传输目标选择 (TS)、功率控制 (PC) 和功率分配 (PA) 表示。更详细地说，TS 问题通过深度 Q 学习 (DQL) 来解决，以减少 BR(也减少了一个无用的中继)，而 PC 和 PA 问题通过 DDPG 方法解决，该方法允许匹配流量需求与合适的传输功率。尽管每个无人机以分布式行为执行这三个模块中的每一个，但值得注意的是，实现了 U2U 通信，因此，邻居无人机在接收到一跳邻居代理的本地观察后，可以相应地进行通信和协作: 使用图论，我们可以将一跳邻居定义为靠近源节点和目的节点之间特定路径的节点 (即，代理)。Cheng 等人 [48] 将他们的技术称为 MAHDRL，它具有与 QMIX 算法相似的结构，但使用混合动作空间和演员-批评家结构来处理它: 在这里解决了联合无人机-用户关联、信道分配和传输功率控制 (J-UACAPC) 问题。使用 MAHDRL 方法背后的主要思想是识别产生最大动作值的混合动作对 (即，离散-连续动作对)。Zhu 等人 [49] 开发了一种分解的 MADDPG(DE-MADDPG) 算法，他们研究了一个涉及无人机集群的场景，通过重建一个连接网络，提出了一个虚拟导航器来解决状态空间的不稳定问题: 虚拟无人机与无人机集群的重心相关联，因此允许考虑相对坐标而不是绝对坐标。De-MADDPG 将 MADDPG 与使用奖励解耦的单代理 DDPG 相结合，即全局奖励和局部奖励。在这里，无人机被表示为图节点，其边表示两个代理之间的通信链接，并取决于每个代理之间的距离。Zhou 等人 [50] 使用一种混合 DRL 方法和以 QoE 为驱动的自适应部署策略来建模他们的问题: 在目标区域的特定位置部署多个无人机，通过无线连接为移动地面用户 (GUs) 提供服务。每个代理被提供一个值网络 (分布式控制)，定义为深度递归 Q 网络 (DRQN)，而一个混合网络 (是一个前馈 NN) 被用来传递来自每个代理的输出到一个旨在进行决策过程的最后一步的层 (集中训练)。为了加快混合 DRL 训练阶段，初始无人机部署通过遗传算法 (GA) 执行，即基于启发式搜索的进化理论启发的算法 (参见 Katoch 等人 [61] 的综述以获取更多详细信息)；通过确保每个无人机之间的连通性和避碰来实现合作。同样的场景在略有不同的目标 (例如，覆盖最大化而不是 QoE 最大化) 已经被 Ma 等人 [62] 处理，并由 Wu 等人 [63] 进一步扩展，后者还使用了注意力机制。考虑到最后

三篇引用的论文，只有一篇 [50] 将被插入到比较表 4 中，因为它提供了关于所使用学习结构的最为详细和清晰的信息，而其他的研究要么是不够详尽的工作 [62]，要么是简单的扩展，除了在 HDRL 技术中使用注意力机制外没有相关的变化 [63]。Zhang 等人 [51] 的研究则是一个特殊情况，其中也考虑了一些覆盖约束，但正如本节开头所解释的，它们不足以让我们将其归类为覆盖类别。因此，作者提出了一种去中心化范式 (实际上，无人机作为独立代理行动) 与 PS 和受限 DQN(cDQN) 算法，以设计无人机轨迹，这些轨迹允许满足一些覆盖约束，只是为了提高通信系统的容量，其中地面终端 (GTs) 被认为是由代理覆盖的。

The primary control and non-payload communication (CNPC) [64] among UAVs is used as a communication model among agents, and thus, each UAV can access the local positions of all the other UAVs. In some cases, the main goal could be trying to minimize the Age of Information (AoI) using UAVs [54], where drones perform different sensing tasks while improving the AoI. In this case, an algorithm called Compound-action Actor-Critic (CA2C) is used: it is based on the DQN and DDPG algorithms to handle actions with discrete and continuous variables. A distributed sense-and-send protocol is used with an actor-critic structure defined for each UAV, and cooperation among agents is obtained through shared reward information collected by a BS and then sent to all the UAVs. The 3D case is tested only with two UAVs, and since we are considering only UAV systems made up of a number of UAVs equal to or greater than three, the third dimension will not be considered as enabled for this work. The work presented by Hu et al. (2020) [54] seems to be an extension of another work [65]. Indeed, three authors out of five are the same: the main difference between these two studies is basically given by the fact that in the former, a more extensive comparative analysis is performed by varying, for example, the agents' flight altitude for different use-cases. For this reason, we consider only the study by Hu et al. (2020) [54] in Table 4. The AoI problem was also investigated by Wu et al. [56], who applied a DDPG-based algorithm. The main difference with respect to the classical DDPG approach is that, here, the training sample is generated after each cycle (including all its samples) and not after each state transition: a cycle is made up of temporal steps depending on some specific triggering events. An experience replay is also used in order to generate the training set. A BS here allows each UAV to know the other agents' state. This paper is in an extension of another work by Wu et al. (2020) [66], and for this reason, the latter will not be taken into account in Table 4, as it does not add/modify any feature to the work by Wu et al. [56]. Similar reasoning applies to the study by Wu et al. (2019) [67], which is basically a less extensive version of the already mentioned manuscript by Wu et al. (2020) [53]. It is worth noting also that the same four authors' names appear in all the latest four cited works (except for two cases in which other two authors are also involved): these works are indeed strictly related as they deal with the same scenario features by either changing the optimization problem (and the used algorithm accordingly) or including some slightly different features and/or baselines algorithms. This is why we decided to insert in the comparative Table 4 only the more exhaustive articles in terms of features and comparative analysis. Some other works can still be mentioned as communication is the application class with the largest number of studies. For example, Chen et al. [57] used UAVs as base stations to provide ground user coverage. The agents perceived partial information about the positions of the M nearest ground users and the position of all the UAVs. Those authors proposed a decentralized trajectory and power control (DTPC) algorithm based on MADDPG, but instead of using a different critic for each agent, they used a single centralized critic. A comparative analysis with multiple baselines such as Dueling DQN, Double DQN, DDPG, and MADDPG was also performed. Heterogeneity in the tasks of the UAVs can be found in some specific applications [60], where a cooperative jamming problem was studied by letting UAV jammers help UAV transmitters defend against ground eavesdroppers (GEs). The authors proposed an improved version of MADDPG called Continuous Action Attention MADDPG (CAA-MADDPG). In particular, the attention mechanism was used to give proper weights to the position of GUs (transmitter UAV case) and GEs (jammer UAVs case). The study presented by Zhang et al. [60] is an extension of another less representative work [68], where MADDPG was applied to the same use-case scenario. For this reason, the latter will not be included in the comparison Table 4. Finally, Zhou et al. [59] investigated a UAV-assisted communication scenario by using a UAV-assisted fair communication (UAFC) algorithm: basically, a multi-agent twin delayed deep deterministic policy gradient (MATD3) algorithm was applied with an actor-critic structure and a CTDE paradigm. In particular, the gates functions allow the agents to select the required information from a central memory according to their observations. In addition, the actor network of the UAVs involves a decomposing and coupling structure: it can decouple the input vector into three categories and expand and aggregate three parts of the information as a unique vector, which helps reduce the state dimension and generate a better policy. The authors also defined a metric to maximize fair system throughput and to guarantee user fairness simultaneously.

无人机之间的主要控制和非载荷通信 (CNPC)[64] 被用作代理之间的通信模型，因此，每个无人机都

可以访问其他所有无人机的本地位置。在某些情况下，主要目标可能是尝试使用无人机最小化信息年龄 (AoI)[54]，在这种情况下，无人机执行不同的感知任务，同时提高 AoI。在这种情况下，使用了名为复合动作演员-评价者 (CA2C) 的算法: 它基于 DQN 和 DDPG 算法来处理具有离散和连续变量的动作。使用分布式感知-发送协议，并为每个无人机定义了演员-评价者结构，通过由基站 (BS) 收集并随后发送给所有无人机的共享奖励信息来实现代理之间的协作。仅对两个无人机的 3D 案例进行了测试，由于我们只考虑由三个或更多无人机组成的无人机系统，因此第三维度将不为本工作启用。Hu 等人 (2020)[54] 呈现的工作似乎是对另一项工作 [65] 的扩展。实际上，五名作者中有三名是相同的: 这两项研究的主要区别在于，前者通过改变例如代理的飞行高度来对不同用例进行更广泛的比较分析。因此，我们在表 4 中只考虑了 Hu 等人 (2020)[54] 的研究。Wu 等人 [56] 也研究了 AoI 问题，并应用了基于 DDPG 的算法。与经典 DDPG 方法的主要区别在于，这里的训练样本是在每个周期 (包括其所有样本) 之后生成的，而不是在每个状态转换之后: 一个周期由依赖于某些特定触发事件的时间步骤组成。还使用了经验回放来生成训练集。这里的基站允许每个无人机了解其他代理的状态。本文是 Wu 等人 (2020)[66] 的另一项工作的扩展，因此，后者将不会在表 4 中考虑，因为它没有为 Wu 等人 [56] 的工作添加/修改任何功能。类似推理适用于 Wu 等人 (2019)[67] 的研究，这基本上是前面提到的 Wu 等人 (2020)[53] 手稿的简化版本。值得注意的是，除了两个案例中涉及另外两个作者外，所有四个最新引用作品中都出现了相同的四个作者的名字: 这些作品确实紧密相关，因为它们处理相同的场景特征，要么通过改变优化问题 (以及相应使用的算法)，要么包括一些略有不同的特征和/或基线算法。这就是为什么我们决定在比较表 4 中仅插入功能和分析更全面的文章。还有一些其他作品值得一提，因为通信是研究数量最多的应用类别。例如，Chen 等人 [57] 使用无人机作为基站来提供地面用户覆盖。代理获得了关于 M 个最近地面用户的位置的部分信息和所有无人机的位置。这些作者提出了一种基于 MADDPG 的分布式轨迹和功率控制 (DTPC) 算法，但不是为每个代理使用不同的评价者，而是使用了一个集中式的评价者。还进行了与 Dueling DQN、Double DQN、DDPG 和 MADDPG 等多个基线的比较分析。在某些特定应用 [60] 中可以找到无人机任务的异质性，其中研究了让无人机干扰者帮助无人机发射者防御地面窃听者 (GEs) 的协作干扰问题。作者提出了 MADDPG 的改进版本，称为连续动作注意力 MADDPG(CAA-MADDPG)。特别是，注意力机制用于为地面用户 (发射无人机案例) 和地面窃听者 (干扰无人机案例) 的位置赋予适当的权重。Zhang 等人 [60] 呈现的研究是另一项不那么具有代表性的工作 [68] 的扩展，其中 MADDPG 应用于相同的用例场景。因此，后者将不包括在比较表 4 中。最后，Zhou 等人 [59] 通过使用无人机辅助公平通信 (UAFC) 算法研究了无人机辅助通信场景: 基本上，应用了具有演员-评价者结构和 CTDE 范式的多代理延迟深度确定性策略梯度 (MATD3) 算法。特别是，门控函数允许代理根据其观察结果从中心存储中选择所需的信息。此外，无人机的演员网络涉及分解和耦合结构: 它可以将输入向量分解为三类，并将信息的三部分扩展和聚合为一个唯一向量，这有助于降低状态维度并生成更好的策略。作者还定义了一个指标，以最大化公平系统吞吐量并同时保证用户公平性。

Other additional final considerations can eventually be taken into account. The number of agents used for communication scenarios is similar among all the works, and no one used more than 15 UAVs. U2U communication is handled only in three studies [47,49,51], while propulsion and communication energy consumption were both considered in the same problem only in one work in the communication class [57]; the 3D scenario is seldomly used except for few cases [49, 51, 52, 60] , while the agents' trajectories are mostly learned but not in three of the analyzed situations [47, 48, 55] , where trajectories are set by default. The collision avoidance task is considered only in about one-third of the works belonging to the communication class [46,50,52,57,58], even if in approximately four-fifths of the works [46,49-54,56-60], the trajectories are learned by agents, and thus, they should try to avoid collisions among other UAVs and/or obstacles.

其他额外的最终考量最终可以被纳入考虑。在所有作品中，用于通信场景的代理数量相似，且没有人使用超过 15 架无人机。仅在三项研究中处理了 U2U 通信 [47,49,51]，而在通信类别中，推进和通信能耗在同一个问题中只在一项工作中被考虑 [57]；3D 场景很少被使用，除了少数情况 [49, 51, 52, 60] ，而代理的轨迹大多数是学习得来的，但在三种分析情况中并未如此 [47, 48, 55] ，其中轨迹是默认设置的。在通信类别中，大约有三分之一的作品考虑了避障任务 [46,50,52,57,58]，尽管在大约五分之四的作品中 [46,49-54,56-60]，轨迹是由代理学习的，因此，它们应该尝试避免与其他无人机和/或障碍物相撞。

4.5. Target-Driven Navigation

4.5. 目标驱动导航

Multi-UAV systems are often used in scenarios involving static or dynamic targets to reach, spot, or track. Based on the works selected for this review, we decided to divide this class into two main sub-classes, i.e., Target Tracking and Target Reaching.

多无人机系统经常被用于涉及静态或动态目标的场景，以到达、定位或跟踪目标。基于本次综述选定

的作品，我们决定将这一类别分为两个主要子类别，即目标跟踪和目标到达。

4.5.1. Target Tracking

4.5.1. 目标跟踪

Many recent applications are focused on trying to make autonomous systems able to track a specific target for different reasons, which could be for entertainment, academic, or safety and surveillance purposes. The main feature of this subclass is that the targets are dynamic, as they need to be tracked.

许多最近的应用专注于尝试使自主系统能够因不同原因跟踪特定目标，这些原因可能是娱乐、学术或安全监控目的。这个子类的主要特征是目标动态，因为它们需要被跟踪。

In Table 5, all the features associated with this specific class are shown. In particular, the feature defined as multi-target indicates whether multiple targets are taken into account during the tracking task: this means that we will consider as multi-target problems both the ones associating only one target to each different agent and the ones in which every UAV could track more than one target.

在表 5 中，展示了与特定类别相关的所有特征。特别是，定义为多目标的特征表示在跟踪任务中是否考虑多个目标: 这意味着我们将把仅将一个目标关联到每个不同代理的问题和每个无人机可能跟踪多个目标的问题都视为多目标问题。

Table 5. Target tracking class features: The mark ∗ indicates class-dependent features and is thus considered only for this class.

表 5. 目标跟踪类别特征: 标记 ∗ 表示类别相关特征，因此仅针对此类进行考虑。

Ref.Algorithm	Par.	No. of UAVs	Col. Av.	Action Space	State Space	3D	UAV Dyn.	Prop. Eg	Multi- Target *
[69] DRQN-RDDPG	FDSR	3	✓	D	C	✓	✓	X	X
[70]MAC3	CTDE	3	✓	C	C	X	X	X	X
[71]MAAC-R	FDLCSR	5,10,20,50, 100,200,1000	X	D	C	X	X	X	✓
[72]PPO	CTDE	2,5,10,15,20	X	D	H	X	X	X	✓
[73]COM-MADDPG	CTDE	3,4	✓	C	X	X	X	X	X
[74]Fast-MARDPG	CTDE	3,4	✓	C	C	X	X	X	✓

参考文献. 算法	参数	无人机数量	科尔尼航空	行动空间	状态空间	3D	无人机动力学	推导示例	多目标 *
[69] DRQN-RDDPG	FDSR	3	✓	D	C	✓	✓	X	X
[70]MAC3	CTDE	3	✓	C	C	X	X	X	X
[71]MAAC-R	FDLCSR	5,10,20,50, 100,200,1000	X	D	C	X	X	X	✓
[72]PPO	CTDE	2,5,10,15,20	X	D	H	X	X	X	✓
[73]COM-MADDPG	CTDE	3,4	✓	C	X	X	X	X	X
[74]Fast-MARDPG	CTDE	3,4	✓	C	C	X	X	X	✓

All the works belonging to the target tracking class apply different algorithms mainly using a CTDE paradigm (except for two cases [69,71]). Yan et al. [72] applies an approach based on the existing PPO algorithm by examining a search and rescue (SAR) problem. SAR problems are actually defined as a combination of searching and tracking tasks, but as we classified these two tasks into separate categories, we put SAR inside the target tracking class since the tracking part in a SAR mission is less exploratory than the searching task (and thus generally more suitable to an RL approach).

属于目标跟踪类别的所有工作主要采用 CTDE 范式 (除两个案例 [69,71] 外) 应用不同的算法。Yan 等人 [72] 采用了一种基于现有 PPO 算法的方法，通过研究搜索和救援 (SAR) 问题。实际上，SAR 问题被定义为搜索和跟踪任务的组合，但由于我们将这两个任务分类为不同的类别，我们将 SAR 放入目标跟踪类别中，因为 SAR 任务中的跟踪部分比搜索任务 (因此通常更适合强化学习方法) 探索性较小。

Some other works belonging to this class try instead to apply more noticeable variations of already known algorithms. For instance, Goh et al. [69] adopted a recurrent approach in two use cases based, respectively, on DQN and DDPG in a multi-UAV actor-critic system to integrate information over time with a combined reward. A reciprocal reward multi-agent actor-critic (MAAC-R) algorithm was used by Zhou et al. [71], who applied an algorithm based on a multi-agent actor-critic by using the parameter sharing technique: here, each agent received an individual reward and the maximum reciprocal reward, i.e., the dot product of the reward vector of all neighbor agents and the dependency vector between the considered agent and its neighbors. This work is an extension of a minor one [75], where the same multi-target tracking (MTT) problem was addressed by the same authors (except for one of them not involved here) but applying a D3QN approach by modeling local communication through graphs; a lower number of UAVs was involved also with the usage of cartogram feature representation (FR) to integrate variable length information into a fixed-shape input size. Thus, only the work by Zou et al. [71] will be considered in Table 5. Jiang et al. [73] proposed a MADDPG-based algorithm with the usage of communication networks (COM-MADDPG): an actor-critic structure was used and, more in depth, a critic, an actor, and communication networks were used and associated with their corresponding target networks. The communication network, which allows for exchanging information locally among the agents, obviously facilitated the UAVs' cooperation in achieving the desired goal. A Fast-MARDPG approach was instead proposed by Wei et al. [74]. Since the standard MADDPG algorithm can only act on the current state, the authors improved the recurrent deterministic policy gradient (RDPG) algorithm by combining RDPG and MADDPG. Other authors [70] used a method referred to as MAC ^{3} with curriculum learning (CL) and joint state and action tracker (JSAT), where a long short-term memory (LSTM) was applied to estimate global observation and action over time: the considered scenario involves pursuers defined as cellular-connected UAVs and an evader indicated as an unauthorized UAV. CL is a technique associated with the training phase, where complex scenario constraints are progressively introduced in such a way that the agents can gradually learn from a more and more different and challenging environment.

一些属于此类的工作尝试应用已知算法的更明显变体。例如，Goh 等人 [69] 在两个用例中采用了循环方法，分别基于 DQN 和 DDPG 的多无人机演员-评论家系统中，随时间整合奖励。Zhou 等人 [71] 使用了一种基于多演员-评论家的算法，并采用了参数共享技术，称为互惠奖励多代理演员-评论家 (MAAC-R) 算法: 在这里，每个代理获得一个单独的奖励以及最大的互惠奖励，即所有邻居代理的奖励向量和考虑代理与其邻居之间的依赖向量的点积。这项工作是较小作品 [75] 的扩展，其中相同的作者 (除了其中一位未参与本研究) 解决了相同的多目标跟踪 (MTT) 问题，但应用了通过图建模局部通信的 D3QN 方法；还涉及较少的无人机数量，并使用地图特征表示 (FR) 将可变长度信息整合为固定形状的输入大小。因此，表 5 中仅考虑 Zou 等人 [71] 的工作。Jiang 等人 [73] 提出了一种基于 MADDPG 的算法，并使用了通信网络 (COM-MADDPG): 使用了一个演员-评论家结构，更深入地说，一个评论家、一个演员和通信网络被使用并与它们对应的目标网络相关联。通信网络允许代理之间在本地交换信息，显然促进了无人机在实现预期目标时的合作。Wei 等人 [74] 则提出了 Fast-MARDPG 方法。由于标准的 MADDPG 算法只能对当前状态进行操作，作者通过结合 RDPG 和 MADDPG 改进了循环确定性策略梯度 (RDPG) 算法。其他作者 [70] 使用了一种称为 MAC ^{3} 的方法，并结合了课程学习 (CL) 和联合状态与动作跟踪器 (JSAT)，其中应用了长短期记忆 (LSTM) 来估计随时间的全局观察和动作: 所考虑的场景包括定义为蜂窝连接无人机的追踪者，以及表示为未经授权无人机的逃避者。CL 是与训练阶段相关的一种技术，其中复杂场景约束逐渐引入，以便代理能够从越来越不同和具有挑战性的环境中逐渐学习。

None of the studies in Table 5 took into account the energy UAV flight consumption, and a remarkable number of UAVs ranging between 5 and 10,000 was considered only in one of them [71]. An effective three-dimensional scenario seems to be involved only in the use-case described by Goh et al. [69], where a particular task associated with this application class was defined: indeed, here, multiple UAVs were supposed to capture the same scene or target simultaneously while flying for aerial cinematography. It is worth mentioning also that, even if an adversarial scenario is supposed to take place in the problem analyzed by Wei et al. [74], we do not consider this work as belonging to the adversarial search and game scenario class as a real game between different teams is not actually considered, but only target tracking tasks are really dealt with.

表 5 中的研究均未考虑无人机飞行能耗，只有一个研究考虑了数量在 5 到 10,000 架之间的无人机 [71]。一个有效的三维场景似乎只在 Goh 等人 [69] 描述的使用案例中涉及，其中定义了与这类应用相关的一个特定任务: 实际上，在这里，多个无人机被假定在飞行中进行空中摄影时同时捕捉同一场景或目标。值得一提的是，尽管在 Wei 等人 [74] 分析的问题中假定存在对抗性场景，但我们并不认为这项工作属于对抗性搜索和游戏场景类别，因为实际上并没有考虑不同团队之间的真正游戏，而只是处理目标跟踪任务。

4.5.2. Target Reaching

4.5.2. 目标到达

In this specific subclass, the targets are known by the UAVs. They aim to generate a trajectory to reach a specific target while satisfying other constraints, such as collision avoidance.

在这个特定的子类别中，无人机知道目标的位置。它们的目的是生成一条轨迹以到达特定的目标，同时满足其他约束条件，例如避碰。

The CTDE learning paradigm is used in four works [76-79] out of seven, while decentralized training with either local or global communication is adopted in the remaining three papers belonging to this class. Well-known DRL algorithms are mostly used in different studies, but some interesting variations are sometimes applied. A particular case of study is the one associated with the multi-UAV flocking problem [79] faced through a digital twin (DT) DRL-based framework (with an actor-critic structure) reproducing a scenario in which UAVs must reach a target location. The approach here applied is referred to by the authors as a Behavior-Coupling Deep Deterministic Policy Gradient (BCDDPG), and it involves four sub-actor networks. Three of the used sub-networks take as input three different types of state information and output three different types of actions (associated with the related input), while the fourth sub-actor takes as input all the three actions generated by the other three sub-actors and the joint state space information derived from their single state information: the fourth sub-actor is thus able to output a desired final action. This method can help in generating learned policies with higher quality. In addition, one of the three sub-networks fed with the decomposed input uses an RNN (more in detail, an LSTM) to extract better information from the history of the data passed as input. Since energy consumption here is only used as a performance comparison metric and not used in the reward function, we will not consider it in Table 6 (the energy consumption is indeed indirectly computed as it can be associated with the average travel distance of the agents). A similar multi-UAV problem was faced by Zhao et al. [77], who proposed multi-agent joint proximal policy optimization (MAJPPO) using a state-value function (as in PPO) instead of the Q-value function (as in DQN). In order to enhance the cooperation and stability of the training, they used a moving window average of the state-value function between different agents. More in detail, each agent computes its state-value function through its critic network, and then, a weighted average is used to compute a joint value function for each of them. The joint state-value function is used to optimize the actor network, which controls the movement of the agents. The UAV dynamic model is also taken into account. A DDPG-based approach was used instead by Wang et al. [80], where a two-stage reinforcement learning method was used mainly to face the problem of multi-UAV collision avoidance under uncertainties (it is modeled as a POMPD). The first stage uses a supervised training technique to train the agent to follow well-known collision avoidance strategies, while the second stage uses a policy gradient technique to improve the previously learned strategies. Here, a single policy with experiences from all the UAVs was trained and then distributed during testing. Although the authors considered this approach a CTDE scheme, we argue that this is a decentralized approach with parameter sharing since there is not a real centralized network that uses the global state of the environment as the input. Another interesting study [81] can be considered as a sort of target discovery problem. However, since it seems to be the only one almost entirely related to this class type, we decided to associate it with the target searching category. The authors considered a surveillance task with multiple targets to be discovered by a group of UAVs. The scenario is modeled as a network distributed POMDP (ND-POMDP), meaning that not all the agents interact simultaneously: a DDQN algorithm with a parameter-sharing technique was applied. In some cases [76,78,82], a planning approach, and, more in detail, a path planning method, was combined with the DRL technique. Indeed, Qie et al. [76] proposed a solution based on multi-agent deep deterministic policy gradient (MADDPG) to solve multi-UAV target assignment and path planning problems simultaneously: the authors proposed STAPP, i.e., a Simultaneous Target Assignment and Path Planning algorithm. In this problem, the UAV must cover a target while cooperating to minimize the total travel cost. The goal of the target assignment problem was to minimize the traveling distance of UAVs, while path planning should minimize collisions. A comparative analysis with respect to other algorithms was not performed, but many experiments were executed and the results were reported by following some metrics such as Collision Rate Between Agent and Agent (CRBAA), Collision Rate Between Agent and Threat area (CRBAT), and the task completion rate. Lin et al. [78] also combined path planning with a DRL approach: MADDPG was applied to avoid collisions between agents, while path planning helped avoid collisions between agents and static obstacles by providing the agents with info about forbidden areas. A global planner was used instead by Walker et al. [82]. A target-reaching problem was investigated, where a planner coordinated different UAVs to reach some target location in a decentralized way. First, the planner collected local observations from the agents and sent them macro-actions indicating the path to achieve the desired location. Then, UAVs followed the global map information received from the global planner by communicating with each other some local observations when needed: each agent was provided with a local controller based on the PPO algorithm to tune the velocity of the considered UAV properly.

在七篇作品中，有四篇采用了 CTDE 学习范式 [76-79]，而其余三篇则采用了局部或全局通信的分布式训练。知名强化学习算法在各项研究中被广泛使用，但有时也会应用一些有趣的变体。一个特殊的研究案例是涉及多无人机群集问题 [79]，通过基于数字孪生 (DT) 的强化学习框架 (具有演员-评论家结构) 来重现无人机必须到达目标位置的场景。作者将此处应用的方法称为行为耦合深度确定性策略梯度 (BCDDPG)，它涉及四个子演员网络。其中三个子网络接收三种不同类型的状态信息作为输入，输出三种不同类型的动作 (与相关输入相关)，而第四个子演员接收其他三个子演员生成的三种动作和从它们单个状态信息中得出的联合状态空间信息作为输入: 因此，第四个子演员能够输出期望的最终动作。这种方法可以帮助生成质量更高的学习策略。此外，接收分解输入信息的三个子网络之一使用 RNN(更具体地说，是 LSTM) 从输入数据的历史中提取更多信息。由于在此处能量消耗仅作为性能比较指标，并未在奖

励函数中使用，因此我们将在表 6 中不考虑它 (能量消耗确实可以与代理的平均旅行距离相关联，因此是间接计算的)。Zhao 等人 [77] 面临了一个类似的多无人机问题，他们提出了基于状态值函数的多智能体联合近似策略优化 (MAJPPO)(与 PPO 中的状态值函数相同)，而不是 DQN 中的 Q 值函数。为了增强训练的合作性和稳定性，他们使用了不同智能体之间的状态值函数的移动窗口平均值。更具体地说，每个智能体通过其评论家网络计算其状态值函数，然后使用加权平均计算它们的联合价值函数。联合状态值函数用于优化控制智能体移动的演员网络。无人机的动态模型也被考虑在内。Wang 等人 [80] 则采用了基于 DDPG 的方法，其中主要使用了两阶段强化学习法来应对不确定情况下的多无人机避障问题 (它被建模为 POMPD)。第一阶段使用监督训练技术训练智能体遵循已知的避障策略，而第二阶段使用策略梯度技术改进之前学习的策略。在这里，从所有无人机的经验中训练了一个单一策略，然后在测试过程中进行分发。尽管作者认为这种方法是 CTDE 方案，但我们认为这是一个带有参数共享的分布式方法，因为没有真正的集中式网络使用环境的全局状态作为输入。另一个有趣的研究 [81] 可以被视为一种目标发现问题。然而，由于它似乎是与这类类型几乎完全相关的唯一研究，我们决定将其归类为目标搜索类别。作者考虑了一个由多无人机发现的多个目标的监控任务。该场景被建模为分布式 POMDP(ND-POMDP)，意味着并非所有智能体同时交互: 应用了带有参数共享技术的 DDQN 算法。在某些情况下 [76,78,82]，一种规划方法，更具体地说，是一种路径规划方法，与强化学习技术相结合。事实上，Qie 等人 [76] 提出了一种基于多智能体深度确定性策略梯度 (MADDPG) 的解决方案，以同时解决多无人机目标分配和路径规划问题: 作者提出了 STAPP，即同时目标分配和路径规划算法。在这个问题中，无人机必须在合作的同时覆盖目标，以最小化总旅行成本。目标分配问题的目标是最小化无人机的旅行距离，而路径规划应最小化碰撞。没有与其他算法的比较分析，但进行了许多实验，并按照一些指标 (如智能体之间的碰撞率 (CRBAA)、智能体与威胁区域的碰撞率 (CRBAT) 和任务完成率) 报告了结果。Lin 等人 [78] 也将路径规划与强化学习方法相结合:MADDPG 应用于避免智能体之间的碰撞，而路径规划通过为智能体提供禁止区域的信息来帮助避免智能体与静态障碍物的碰撞。Walker 等人 [82] 则使用了全局规划器。研究了一个目标到达问题，其中规划器以分布式方式协调不同无人机到达某个目标位置。首先，规划器收集智能体的本地观察结果，并发送表示达到期望位置路径的宏动作。然后，无人机通过相互通信在某些情况下需要时发送一些本地观察结果，遵循从全局规划器接收的全局地图信息: 每个智能体都配备了一个基于 PPO 算法的本地控制器，以适当调整考虑的无人机的速度。

Table 6. Target-reaching class features:the mark ∗ indicates class-dependent features and is thus considered only for this class.

表 6. 目标到达类特征: 标记 ∗ 表示类相关特征，因此仅针对此类进行考虑。

Ref.	Algorithm	Par.	No. of UAVs	Col. Av.	Action Space	State Space	3D	UAV Dyn.	Prop. Eg	Multi- Target *
[76]	STAPP	CTDE	4,5	✓	X	X	✓	X	X	✓
[77]	MAJPPO	CTDE	3	✓	C	C	✓	✓	X	X
[80]	DDPG	FDLC	10,200	✓	X	X	✓	✓	X	X
[82]	PPO	FDGC	1–4	✓	C	C	✓	✓	X	✓
[78]	MADDPG	CTDE	3	✓	D	C	X	X	X	✓
	BCDDPG	CTDE	6.9.12	✓	C	C	X	X	X	X
	DDQN	FDGC	2,3,4	✓	D	D	X	X	X	✓

参考文献	算法	参数	无人机数量	列平均	动作空间	状态空间	3D	无人机动力学	推力示例	多目标 *
[76]	STAPP	CTDE	4,5	✓	X	X	✓	X	X	✓
[77]	MAJPPO	CTDE	3	✓	C	C	✓	✓	X	X
[80]	DDPG	FDLC	10,200	✓	X	X	✓	✓	X	X
[82]	PPO	FDGC	1–4	✓	C	C	✓	✓	X	✓
[78]	多智能体深度确定性策略梯度 (MADDPG)	CTDE	3	✓	D	C	X	X	X	✓
	双曲线正切深度确定性策略梯度 (BCDDPG)	CTDE	6.9.12	✓	C	C	X	X	X	X
	双重深度 Q 网络 (DDQN)	FDGC	2,3,4	✓	D	D	X	X	X	✓

We noticed that energy consumption is never considered in this specific class, but the collision avoidance task is always taken into account. The three-dimensional use-case is dealt with in approximately half of the works considered here [76, 77, 80, 82] and a significantly larger number of UAVs (up to 200) were involved only in one of them [80].

我们注意到，在这个特定类别中，从未考虑过能耗，但总是考虑到避障任务。在本文考虑的约一半作品中处理了三维用例 [76, 77, 80, 82] ，并且仅在其中一个作品中涉及了数量显著更多的无人机 (多达 200 架)[80]。

5. Discussion

5. 讨论

From Section 4, it is possible to infer relevant considerations and significant aspects related to the DRL-based multi-UAV systems.

从第 4 节中，可以推断出与基于 DRL 的多无人机系统相关的考虑和重要方面。

5.1. Technical Considerations

5.1. 技术考虑

We can genuinely state that whenever it is possible to assume that the static objects of the environment are known, a DRL model-based approach must be taken. Indeed, a model-free approach could be reasonable only in cases where UAVs operate in a completely unknown environment, such as a cave or a catastrophic scenario, which were mostly considered in SAR missions. Nevertheless, only a few works seem to use model-based techniques (e.g., [76,78,82]). It should also be noted that none of the selected papers included the delay in the learning process. In a delayed MDP framework [83], we could have three different types of delay:

我们可以真诚地说，在可以假定环境中的静态对象已知的情况下，必须采用基于 DRL 模型的策略。实际上，只有在无人机在完全未知的环境中运行，例如洞穴或灾难场景中，模型自由的方法才可能是合理的，这些情况在 SAR 任务中主要被考虑。然而，似乎只有少数作品使用了基于模型的技巧 (例如 [76,78,82])。还应注意的是，所选论文中没有一篇将延迟包含在学习过程中。在延迟 MDP 框架 [83] 中，可能有三种不同类型的延迟:

- Reward delay. The reward of the agents can be received only after a specific time interval [84, 85] due, for example, to some unpredictable constraints of the considered multi-agent system;

- 奖励延迟。由于考虑的多代理系统中存在一些不可预测的约束，代理的奖励可能只有在特定时间间隔后才能接收到 [84, 85] ;

- Observation delay. Some observations can be delayed (e.g., [86]), and this may occur in real multi-UAV systems either for sensing or communication reasons;

- 观测延迟。某些观测可能会延迟 (例如，[86])，并且这可能会在实际的多无人机系统中由于感知或通信原因而发生;

- Action delay. In multi-UAV applications, some actions could take some time before being accomplished, resulting in a system affected by delayed actions [87].

- 动作延迟。在多无人机应用中，某些动作可能需要一些时间才能完成，导致系统受到延迟动作的影响 [87]。

Considering one, some, or all of the delays mentioned above can ease the knowledge transferability from a simulated environment to a real one. This transfer can also be facilitated by the usage of realistic UAV dynamic models involving low-level input quantities (such as forces and torques) controlling the UAVs. Only a few works [69,77,80,82] use this likely UAVs' representation and only through simulators in target-driven navigation applications. Some other and more technical observations can be made for what concerns the MDP and its main features (e.g., state space, action space, and reward function), which need to be described clearly and concisely and possibly defined by using the AI field terminology in order to avoid any kind of ambiguity and misunderstanding (e.g., in the work by Patrizi et al. [88], the MDP features are not clearly described and neither a pseudo-code nor a flowchart of the algorithm used is provided). In addition, we want to remark on using some particular learning paradigms associated with a sequential training phase [89], where during the training of the agent k , all the learned policies of the other agents are kept fixed. Proceeding in this way, the problem of the non-stationarity of the environment will still be present at execution time, and thus, it should be avoided real multi-UAV application deployment. It is worth mentioning that authors tend to implement custom scenarios for their considered use case: agreeing on a universal framework (or on a limited group of frameworks) is needed to perform extensive and fair comparative analyses in the most modular and scalable way possible. Moreover, the various comparison tables in Section 4 show that a 3D environment is not very often considered in multi-UAV systems. This choice could be a valid one regardless of whether it is always possible to perform a UAV flight level assignment, but in extremely dynamic scenarios (e.g., emergency or catastrophic), it was not possible to allocate the UAVs in a specific-altitude space slot both at the right time and safely. Thus, in all these cases, UAVs need to learn a policy that can be performed throughout three-dimensional space. Another remarkable aspect is that authors mainly focus on DRL-based baselines when performing a comparative analysis with their proposed approach. Researchers should try instead to also use baseline algorithms that are not based on DRL in order to perform a fairer comparison analysis: in this way, they could highlight more markedly and clearly the possible effectiveness of a DRL-based approach against already existing deterministic and classical methods.

考虑上述提到的一个、一些或所有延迟，可以减轻从模拟环境到实际环境的知识迁移性。这种迁移还可以通过使用涉及低级别输入量 (如力和扭矩) 的真实无人机动态模型来促进，这些输入量控制着无人机。只有少数研究工作 [69,77,80,82] 使用了这种可能的无人机表示，并且仅在以目标驱动导航应用中通过模拟器进行。关于 MDP 及其主要特征 (例如，状态空间、动作空间和奖励函数)，还可以做出一些更技术性的观察，这些特征需要清晰简洁地描述，并可能使用 AI 领域的术语来定义，以避免任何类型的模糊和误解 (例如，在 Patrizi 等人 [88] 的工作中，MDP 特征没有明确描述，也没有提供算法使用的伪代码或流程图)。此外，我们要强调使用某些与顺序训练阶段 [89] 相关的特定学习范式，在训练代理 k 期间，其他代理学习的所有策略都保持不变。以这种方式进行，环境非平稳性的问题在执行时仍然存在，因此，在真实的无人机应用部署中应避免这种情况。值得注意的是，作者倾向于为他们的使用场景实现自定义场景: 需要就一个通用框架 (或一组有限的框架) 达成一致，以便以最模块化和可扩展的方式执行广泛且公正的比较分析。此外，第 4 节中的各种比较表格显示，在多无人机系统中很少考虑 3D 环境。这个选择可能是有效的，无论是否总是能够执行无人机飞行级别分配，但在极端动态场景 (例如紧急或灾难性情况) 中，无法在正确的时间和安全地分配无人机到特定的空域高度槽。因此，在这些所有情况下，无人机需要学习可以在三维空间中执行的政策。另一个值得注意的方面是，作者在与其提出的方法进行对比分析时，主要关注基于 DRL 的基线。研究人员应该尝试也使用非基于 DRL 的基线算法，以进行更公平的比较分析: 这样，他们可以更明显、更清晰地突出基于 DRL 的方法相对于现有的确定性方法和经典方法可能的功效。

Finally, a straightforward overview of the DRL approaches used in multi-UAV systems is given by the scheme shown in Figure 4. Inside the additional features set shown in this Figure, attention means the application of an attention mechanism to any neural network involved in the usage of a DRL algorithm, while hybrid represents a technique either mixing different DRL approaches or including also deep learning methods. This scheme is quite straightforward, and hence, we can infer the following:

最后，图 4 所示的方案提供了一个关于多无人机系统中所使用深度强化学习 (DRL) 方法的直接概述。在该图所示的附加特性集中，注意力表示对使用 DRL 算法的任何神经网络应用注意力机制，而混合则代表一种技术，它要么混合不同的 DRL 方法，要么包括深度学习方法。这个方案非常直接，因此，我们可以推断以下内容:

- DDPG and PPO are the most used policy-based algorithms;
- DDPG 和 PPO 是使用最广泛的基于策略的算法；
- The most used value-based algorithm is DQN;
- 使用最广泛的基于价值的算法是 DQN；
- MADDPG, which is off policy and mainly associated with a continuous action space, is instead the unique algorithm used in all the macro multi-UAV applications;
- MADDPG 是一种异策略算法，主要与连续动作空间相关联，是在所有宏观多无人机应用中唯一使用的算法；
- The most used techniques combined with DRL methods are the others, namely all methods varying the reward and/or gradient sharing and processing, and more generally not included in all the other additional techniques specified and shown in Figure 4;
- 与 DRL 方法结合使用最广泛的技术是其他技术，即所有改变奖励和/或梯度共享和处理的方法，更一般地，不包括图 4 中指定和显示的所有其他附加技术；

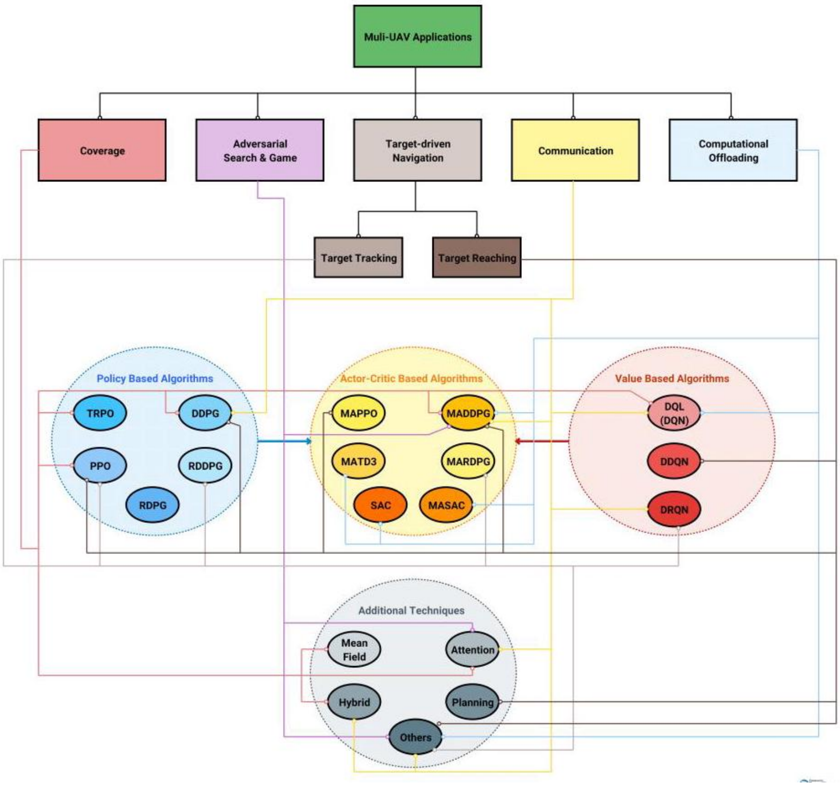


Figure 4. Multi-UAV applications and related DRL algorithms taxonomy. See Section 2.3 for more details on policy-based, value-based and actor-critic algorithms; for a better comprehension of the additional techniques, see instead Section 2.5.

图 4. 多无人机应用及相关 DRL 算法分类。有关基于策略、基于价值和演员-评论家算法的更多细节，请参见第 2.3 节；为了更好地理解附加技术，请参见第 2.5 节。

5.2. Driving Works and Links Analysis

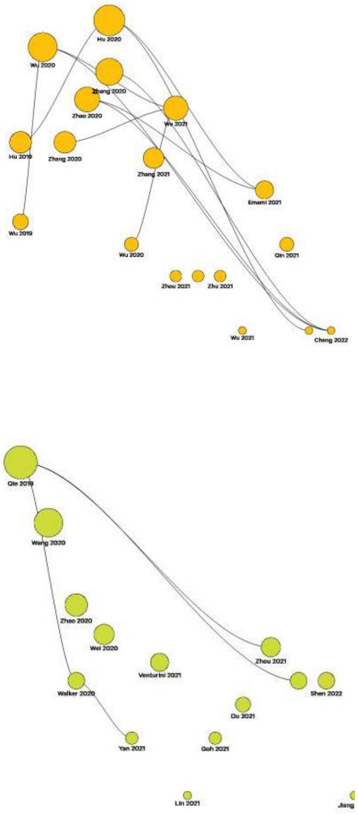
5.2. 驱动作品与关联分析

Here, an overall and visual comparison among the selected papers is conducted in order to understand better whether and possibly how they are linked based either on the specific UAV application considered or on some other extra feature (Figures 5-7 are screenshots taken from interactive maps generated through Litmaps [90]).

在这里，我们对选定的论文进行了总体和视觉上的比较，以更好地理解它们是否以及可能是如何基于特定的无人机应用或某些其他额外特性相互关联的 (图 5-7 是从通过 Litmaps [90] 生成的互动地图中截取的屏幕截图)。

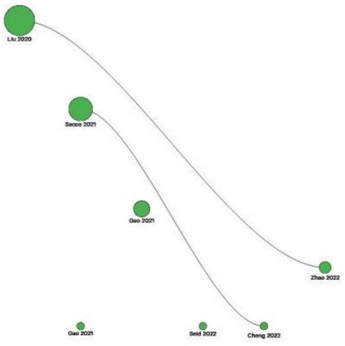
From Figure 5, we can notice that communication is the UAV application related to the most works and associations; target-driven navigation and coverage also present some connections among the papers belonging to the same class, but not as significant as the ones in communication application. Very few arches link instead the works associated with the computational offloading class, while the ones belonging to the adversarial search and game are not connected at all. A particular aspect to be noticed is that even though some of the selected studies have several citations (in absolute terms), they are not taken into account by the papers included in this review. Indeed, Figure 6 shows that some nodes associated with the most cited works (e.g., [24,40,80]) have few or no arches towards the other articles selected in our work: this aspect is quite unexpected as these studies are among the first ones (between all those here included) to apply DRL techniques to cooperative multi-UAV systems, and hence, they should have been analyzed at least by the authors dealing with the same application class. We can further note that the most citations among the papers present in this review are achieved by studies presented by Qie et al. [76] and Wang et al. [58] belonging to the target reaching and coverage classes, respectively, and thus not belonging to the first application class when it comes to the number of works and paper connections, i.e., the communication. Indeed, Figure 6 shows that these works are also mentioned in studies that do not deal with the same UAV application class.

从图 5 中，我们可以注意到通信是与最多作品和研究关联的无人机应用；目标驱动导航和覆盖在属于同一类的论文之间也存在一些联系，但不如通信应用中的联系显著。相反，与计算卸载类相关的工作之间几乎没有关联，而属于对抗搜索和游戏的则完全没有连接。一个值得注意的特殊方面是，尽管一些选定的研究有多篇引用 (从绝对数量上看)，但它们并没有被本综述中包含的论文所考虑。实际上，图 6 显示，一些与被引用次数最多的作品相关联的节点 (例如 [24,40,80]) 很少有或没有指向我们工作中选定的其他文章的弓形: 这个方面非常出乎意料，因为这些研究是第一批 (在所有这里包含的研究中) 将深度强化学习技术应用于协同多无人机系统的，因此，它们至少应该被处理相同应用类的作者分析过。我们还可以进一步注意到，本综述中论文中引用次数最多的是 Qie 等人 [76] 和王等人 [58] 的研究，分别属于目标到达和覆盖类别，因此在作品数量和论文联系方面不属于第一个应用类别，即通信。实际上，图 6 显示，这些作品也被那些不处理相同无人机应用类的研究所提及。



(a) Communication class. (b) Target-driven navigation class. Figure 5. Cont.

(a) 通信类别。(b) 目标驱动导航类别。图 5。续。



Computational offloading class.

(e) Adversarial search and game class.

(e) 对抗搜索和游戏类别。

Figure 5. Theanalyzed papers are shown here as nodes through five different maps indicating the UAV application classes: the bigger the node, the larger the absolute number of citations (regardless of its belonging class) of the paper associated with it. On each map, on the left side are shown the oldest papers, while on the right side are shown the most recent ones; on the bottom side are the less cited papers, while on the top side are the most cited ones. In order to avoid overlapping and to improve readability, the names of the articles are not always reported below the corresponding nodes. The maps are ordered from left to right and from top to bottom based on descending order of the number of citation links (i.e., the arches connecting the nodes) among the studies belonging to the class.

图 5. 分析的论文在这里以节点的形式展示，通过五种不同的映射来指示无人机应用类别: 节点越大，与之关联的论文的引用绝对数量越大 (不考虑其所属类别)。在每一张地图上，左侧显示的是较早的论文，而右侧显示的是最新的论文；底部是引用较少的论文，而顶部是引用最多的论文。为了避免重叠并提高可读性，文章的名称并不总是显示在相应节点的下方。这些地图按照从左到右、从上到下的顺序排列，基于属于该类别的论文之间的引用链接数量 (即连接节点的拱门) 的降序排列。

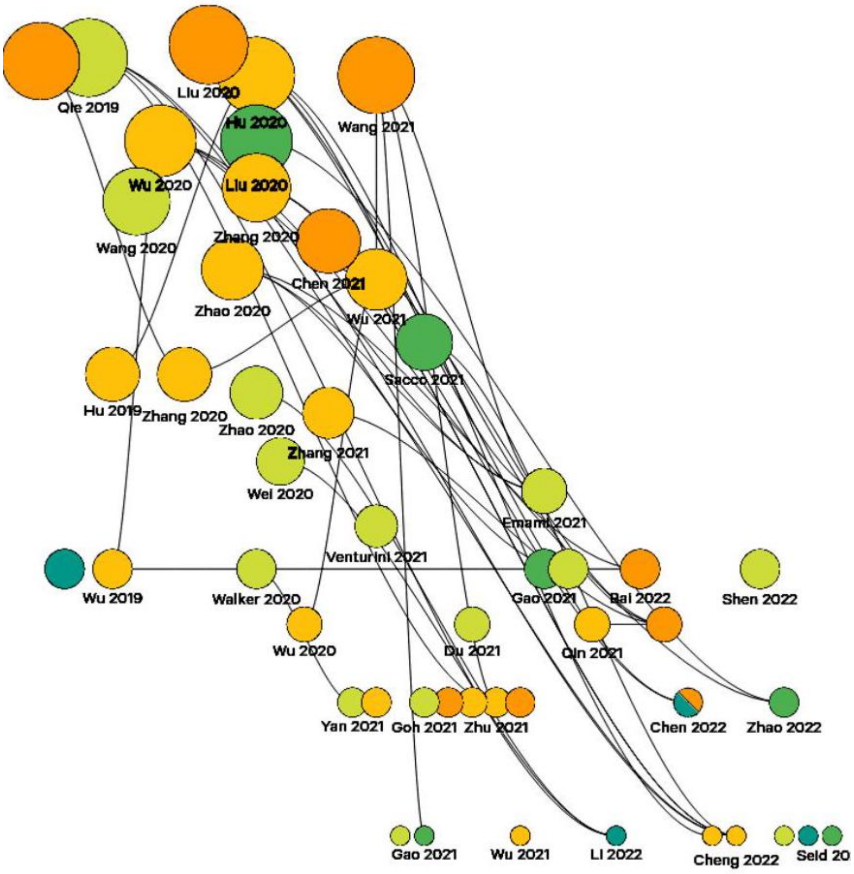


Figure 6. Overall paper comparison: the colors, sizes, and arrangement of the nodes associated with each paper follow the same scheme described in Figure 5.

图 6. 论文总体比较: 与每篇论文关联的节点的颜色、大小和排列遵循与图 5 相同的方案。

A more general overview of the links associated with the papers included in this article is shown in Figure 7, where we can notice that some relevant works were not included in our review (i.e., the black circles placed inside the grey circular crown) as they do not comply with our filtering methodology described in Section 3. Some of them are associated with applications not focused on multi-UAV systems but dealing with wellknown and relevant DRL algorithms such as DQN [7], DDPG [91], or MADDPG [92]. Other ones instead consider UAVs scenarios but without applying (D)RL-based solutions: Zhang et al. (2018) [93] provided a closed-form solution for the joint trajectory design and power control in a UAV relay network, while Zhang et al. (2019) [94] defined a cooperative sense-and-send protocol to improve the quality of service in UAV sensing tasks. Finally, there can also be found all the works related to UAV applications using DRL techniques but without satisfying some specific filters used in our selection methodology: these cases can be represented, for example, by some works in which a centralized learning approach is used [33], or a tabular reinforcement learning strategy is applied instead of a deep one [95].

本文第 7 图中展示了与本文所包含论文相关的更广泛的概述，我们可以注意到，一些相关的工作 (即灰色圆形花环内的黑色圆圈) 并未包含在我们的综述中，因为它们不符合我们在第 3 节中描述的筛选方法。其中一些与不专注于多无人机系统但涉及知名且重要的深度强化学习算法的应用相关，如 DQN [7]、DDPG [91] 或 MADDPG [92]。另一些则考虑了无人机场景，但未应用基于 (深度) 强化学习的解决方案:Zhang 等人 (2018 年)[93] 为无人机中继网络中的联合轨迹设计和功率控制提供了一个闭式解，而 Zhang 等人 (2019 年)[94] 定义了一种协作感知与发送协议，以提高无人机感知任务的服务质量。最后，还可以找到所有使用深度强化学习技术的无人机应用相关研究，但它们不符合我们选择方法中的一些特定筛选条件: 这些情况可以例如表示为使用集中式学习方法的某些研究 [33]，或者应用了表格型强化学习策略而非深度学习策略的研究 [95]。

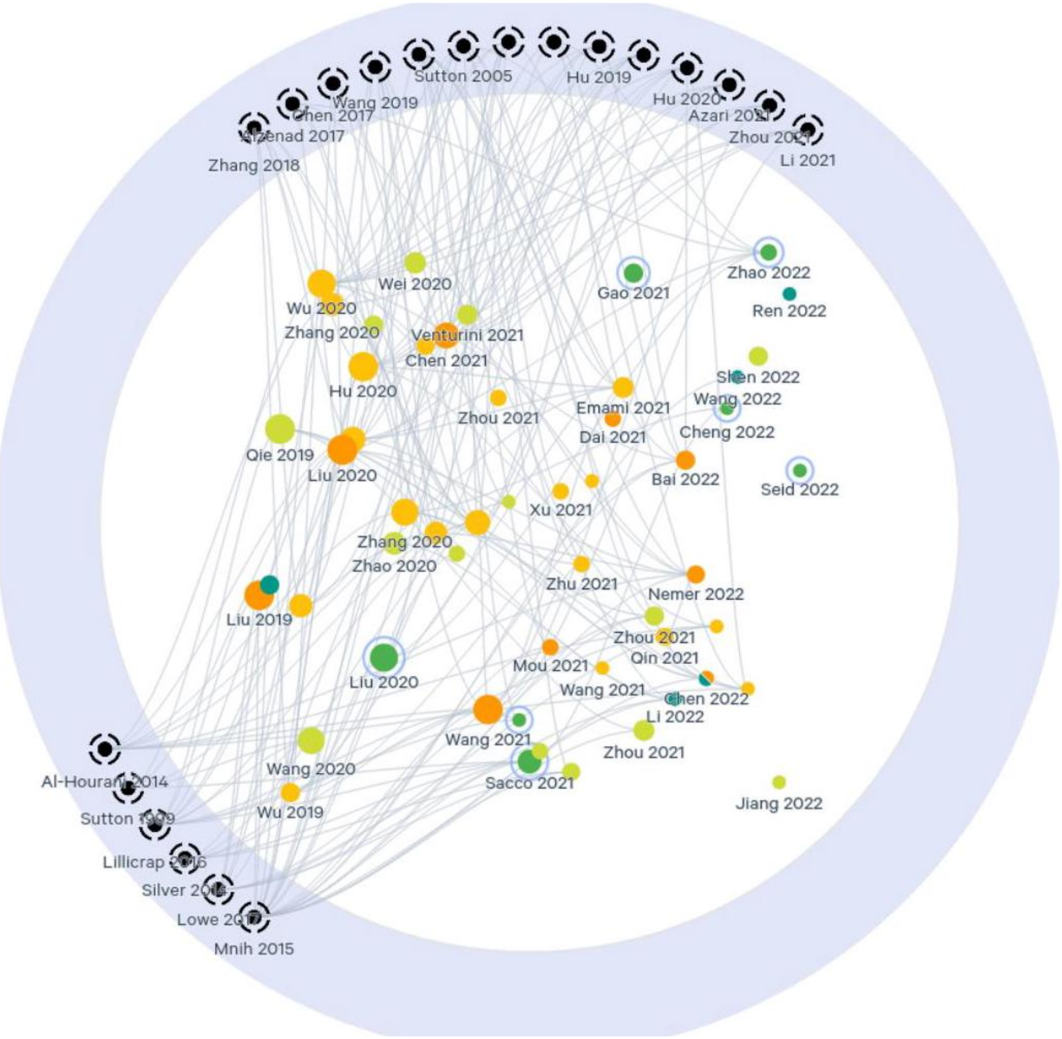


Figure 7. Overview of higher-level papers: the grey circular crown contains the papers not included in this review but which are most cited among those involved in it. The colors, sizes, and arrangement of the nodes associated with each paper follow the scheme described in Figures 5 and 6.

图 7. 高层级论文概述: 灰色圆形花环包含的是未包含在本综述中但在其中被引用最多的论文。每个论文关联的节点的颜色、大小和排列遵循图 5 和图 6 中描述的方案。

5.3. General Considerations on Current and Future Developments

5.3. 关于当前和未来发展的总体考虑

Section 4 allows us to conclude that there exists a preferred direction for what concerns the application scenarios involving multi-UAV DRL-based systems, and it is represented by the communication application class, immediately followed by other classes which are directly or indirectly associated with it, such as the computational offloading and coverage categories. Additionally, target-driven navigation is mainly studied as it involves other subclasses related to target-reaching, target-spotting, and/or target-tracking tasks. The greatest concentration of the works on communication could be explained by the increasing need for a stable, reliable, and highly responsive interconnection and data exchange between devices, not only in urban but also in industrial scenarios. Special attention is deserved by all those multi-UAV applications involving monitoring and surveillance tasks that could result in a limitation of individual privacy and freedom whether or not they are properly designed: both the UAV usage and security aspects of data sensing and exchange (e.g., through the usage of a blockchain [44]) should be investigated more in detail. In this regard, we also highlight the psychological impact that multi-UAV DRL-based systems could have on people who, in their daily lives, are not familiar with either AI or drones. People should be made aware of these future scenarios in which drones will ”fly over their heads” just as cars now run on the roads. Adequate awareness of people on this topic aimed at underlying the relevance of the usage of AI-based multi-UAV systems in crucial applications (e.g., SAR or medical deliveries) could significantly reduce possible demonstrations and/or intolerant attitudes against such future applications.

第 4 节使我们得出结论，关于涉及多无人机深度强化学习系统的应用场景，存在一个首选方向，即通信应用类别，紧接着是与之直接或间接相关的其他类别，例如计算卸载和覆盖类别。此外，目标驱动导航主要被研究，因为它涉及与其他与目标到达、目标定位和/或目标跟踪任务相关的子类别。通信领域作品集中的原因可能是由于在城市场景和工业场景中，设备之间需要一个稳定、可靠且高度响应的互联和数据交换的需求日益增长。特别值得关注的是所有涉及监控和监控任务的多无人机应用，无论它们是否设计得当，都可能限制个人隐私和自由: 无人机使用以及数据感知和交换的安全方面 (例如，通过使用区块链 [44]) 应该更详细地研究。在这方面，我们还强调了基于多无人机深度强化学习系统可能对那些在日常生活中不熟悉 AI 或无人机的人产生的心理影响。人们应该意识到这些未来场景，在这些场景中，无人机将 “在他们的头顶上飞行”，就像现在汽车在道路上行驶一样。人们对这一主题的足够认识，旨在强调基于 AI 的多无人机系统在关键应用 (例如，搜索与救援或医疗配送) 中的使用相关性，这可能会显著减少对这些未来应用的可能的抗议和/或不容忍的态度。

6. Conclusions

6. 结论

As multi-UAV applications and studies in DRL have progressively increased in the last few years, we decided to investigate and discuss the most used DRL techniques for the most emerging multi-UAV use cases in cooperative scenarios. We classified the main multi-UAV applications into five macro-categories: (i) coverage, (ii) adversarial search and game, (iii) computational offloading, (iv) communication, and (v) target-driven navigation. An extensive comparison of all the DRL strategies used for each class has been argued, and improvement considerations have been proposed. DRL is undoubtedly helpful in all cases in which a real-time task execution strategy modification based on dynamic feedback is needed (e.g., learning a new behavior through new and additional knowledge of the environment). For the particular topic covered in this article, system responsiveness is pivotal. Indeed, the eventual need to recompute a valid alternative strategy through default emergency schemes and classical deterministic methods could be heavy and slow in performance and, hence, risky for the system's safety and all the items involved in the considered environment. Thus, DRL results are a winning technique whenever it is needed to make a valid and feasible decision within a strictly limited time interval following a dynamic and/or unpredictable event. These requirements are crucial in multi-UAV application scenarios, and this review provides comprehensive and valuable material and suggestions to face and improve these systems using DRL techniques. In order to satisfy these systems' requirements, we suggest directing future works towards mainly solutions that are not fully centralized but cooperative and that explicitly consider the delay in the algorithm design: the former ensures the system scalability, safety, relatively low computational resources, and the independence from a single central control unit, while the latter eases the simulation-to-real knowledge transferability process.

随着多无人机应用和深度强化学习 (DRL) 研究在过去几年中的逐步增加，我们决定调查和讨论最常用的 DRL 技术，这些技术用于最新兴的多无人机用例中的协作场景。我们将主要的多无人机应用分为五个宏观类别:(i) 覆盖，(ii) 对抗性搜索和游戏，(iii) 计算卸载，(iv) 通信和 (v) 目标驱动导航。对每个类别使用的所有 DRL 策略进行了广泛比较，并提出了改进考虑。在需要根据动态反馈实时修改任务执行策略的所有情况下 (例如，通过新获得的环境知识学习新行为)，DRL 无疑是有所帮助的。对于本文涵盖的特定主题，系统的响应性是关键。实际上，最终需要通过默认的紧急方案和经典的确定性方法重新计算有效替代策略可能是性能沉重且缓慢的，因此对系统的安全性和考虑环境中的所有项目来说都是危险的。因此，在需要在一个严格限定的时间间隔内，根据动态和/或不可预测事件做出有效且可行的决策时，DRL 结果是赢得技术。这些要求在多无人机应用场景中至关重要，本文综述提供了全面且有价值的信息和建议，以应对并改进这些系统，使用 DRL 技术。为了满足这些系统的要求，我们建议将未来的工作主要引导至非完全集中式的解决方案，而是协作性的，并在算法设计中明确考虑延迟: 前者确保了系统的可扩展性、安全性、相对较低的计算资源需求，以及独立于单个中央控制单元，而后者则简化了模拟到现实知识的迁移过程。

Author Contributions: Conceptualization, F.F and D.B; methodology, F.F. and D.B.; investigation, F.F. and D.B.; visualization, F.F. and D.B.; writing-original draft preparation, F.F. and D.B.; writing-review and editing, F.F., D.B. and L.I.; funding acquisition, L.I. All authors have read and agreed to the published version of the manuscript.

作者贡献: 构思，F.F 和 D.B；方法论，F.F. 和 D.B.；调查，F.F. 和 D.B.；可视化，F.F. 和 D.B.；写作-原始草稿准备，F.F. 和 D.B.；写作-审查与编辑，F.F.、D.B. 和 L.I.；资金获取，L.I. 所有作者已阅读并同意发表的稿件版本。

Funding: This research was partially funded by the BUBBLES Project under a grant agreement (No. 893206), by the ERC Advanced Grant WhiteMech (No. 834228), and by the PNRR MUR project PE0000013-FAIR.

资金: 本研究部分得到了 BUBBLES 项目 (资助协议编号 893206)、ERC 高级资助 WhiteMech(编号 834228) 和 PNRR MUR 项目 PE0000013-FAIR 的资助。

Conflicts of Interest: The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

利益冲突: 资助方在研究设计、数据收集、分析或解释、撰写稿件或决定发布结果方面均未发挥作用。

Abbreviations

缩写

Hierarchical Graph Attention
分层图注意力
Multi-Agent Deep Deterministic Policy Gradient
多代理深度确定性策略梯度
Multi-UAV Deep Reinforcement Learning-Based Scheduling Algorithm
基于多无人机深度强化学习的调度算法
Multi-Agent Hybrid Deep Reinforcement Learning
多代理混合深度强化学习
Multi-Agent Joint Proximal Policy Optimization
多代理联合近似策略优化
Multi-Agent Reinforcement Learning Actor-Critic
多代理强化学习演员-评论家
Multi-Agent Soft Actor-Critic
多代理软演员-评论家
Multi-Agent Twin Delayed Deep Deterministic Policy Gradient
多代理双延迟深度确定性策略梯度
RL Terminology and Domain-Dependent Terms
强化学习术语和领域相关术语
Action Sp./Act Sp. Action Space
动作空间
Alg. Algorithm
算法

Base Station
基站

Convolutional Neural Network
卷积神经网络

Collision Avoidance
避碰

Energy
能量

FOMDP Fully Observable Markovian Decision Process
完全可观测马尔可夫决策过程

HJB/FBP Hamilton-Jacobi-Bellman/Fokker-Planck-Kolmogorov
汉密尔顿-雅可比-贝尔曼/福克-普朗克-科尔莫哥洛夫

LSTM Long Short-Term Memory
长短时记忆

MARL Multi-Agent Reinforcement Learning
多智能体强化学习 (MARL)

MCS Mobile Crowd Sensing
移动群体感知 (MCS)

MDP Markovian Decision Process
马尔可夫决策过程 (MDP)

MEC Mobile Edge Computing
移动边缘计算 (MEC)

MFE Mean Field Equilibrium
平均场均衡 (MFE)

MFG Mean Field Game
平均场博弈 (MFG)

MOMDP Mixed Observability Markovian Decision Process
混合可观测马尔可夫决策过程 (MOMDP)

MT Mobile Terminal
移动终端 (MT)

POMDP Partially Observable Markovian Decision Process
部分可观测马尔可夫决策过程 (POMDP)

Par. Paradigm
对比范式 (Par. Paradigm)

Points of Interest
兴趣点 (Points of Interest)

Prop. Eg Propulsion Energy
推进能量命题

Quality of Service
服务质量

RNN Recurrent Neural Network
循环神经网络

Ref. Reference
参考文献

SARL Single-Agent Reinforcement Learning
单代理强化学习

Stochastic Partial Differential Equations
随机偏微分方程

State Sp./St. Sp. State Space
状态空间

U2D UAV-to-Device
无人机到设备的通信

U2U UAV-to-UAV
无人机到无人机的通信

UAV Unmanned Aerial Vehicle
无人驾驶飞行器

UAV Dyn. UAV Dynamic model
无人机动态模型

UE User Equipment
UE 用户设备

References

参考文献

1. Akhloufi, M.A.; Couturier, A.; Castro, N.A. Unmanned Aerial Vehicles for Wildland Fires: Sensing, Perception, Cooperation and Assistance. Drones 2021, 5, 15. [CrossRef]

2. Hayat, S.; Yanmaz, E.; Brown, T.X.; Bettstetter, C. Multi-objective UAV path planning for search and rescue. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May-3 June 2017; pp. 5569-5574. [CrossRef]

3. Aurambout Jean-Philippe, G.K.C.B. Last mile delivery by drones: an estimation of viable market potential and access to citizens across European cities. Eur. Transp. Res. Rev. 2019, 11, 30. [CrossRef]

4. Salhaoui, M.; Guerrero-González, A.; Arioua, M.; Ortiz, F.J.; EI Oualkadi, A.; Torregrosa, C.L. Smart Industrial IoT Monitoring and Control System Based on UAV and Cloud Computing Applied to a Concrete Plant. Sensors 2019, 19, 3316. [CrossRef] [PubMed]

5. Zhou, C.; He, H.; Yang, P.; Lyu, F.; Wu, W.; Cheng, N.; Shen, X. Deep RL-based Trajectory Planning for AoI Minimization in UAV-assisted IoT. In Proceedings of the 2019 11th International Conference on Wireless Communications and Signal Processing (WCSP), Xi'an, China, 23-25 October 2019; pp. 1-6. [CrossRef]

6. Chakareski, J. UAV-IoT for Next Generation Virtual Reality. IEEE Trans. Image Process. 2019, 28, 5977-5990. [CrossRef]

7. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A.A.; Veness, J.; Bellemare, M.G.; Graves, A.; Riedmiller, M.; Fidjeland, A.K.; Ostrovski, G.; et al. Human-level control through deep reinforcement learning. Nature 2015, 518, 529-533. [CrossRef]

8. Silver, D.; Huang, A.; Maddison, C.J.; Guez, A.; Sifre, L.; Van Den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; et al. Mastering the game of Go with deep neural networks and tree search. Nature 2016, 529, 484-489. [CrossRef]

9. Berner, C., Brockman, G., Chan, B., Cheung, V., Debiak, P., Dennison, C., Farhi, D.; Fischer, Q.; Hashme, S.; Hesshme, S.; Hessse, C.; et al. Dota 2 with Large Scale Deep Reinforcement Learning. arXiv 2019, arXiv:1912.06680.

10. OpenAL; Akkaya, I.; Andrychowicz, M.; Chociej, M.; Litwin, M.; McGrew, B.; Petron, A.; Paino, A.; Piaippert, M.; Powell, G.; et al. Solving Rubik's Cube with a Robot Hand. arXiv 2019, arXiv:1910.07113.

11. Bithas, P.S.; Michailidis, E.T.; Nomikos, N.; Vouyioukas, D.; Kanatas, A.G. A Survey on Machine-Learning Techniques for UAV-Based Communications. Sensors 2019, 19, 5170. [CrossRef]

12. Ben Aissa, S.; Ben Letaifa, A. UAV Communications with Machine Learning: Challenges, Applications and Open Issues. Arab. J. Sci. Eng. 2022, 47, 1559-1579. [CrossRef]

13. Puente-Castro, A.; Rivero, D.; Pazos, A.; Fernandez-Blanco, E. A review of artificial intelligence applied to path planning in UAV swarms. Neural Comput. Appl. 2022, 34, 153-170. [CrossRef]

14. Pakrooh, R.; Bohlooli, A. A Survey on Unmanned Aerial Vehicles-Assisted Internet of Things: A Service-Oriented Classification. Wirel. Pers. Commun. 2021, 119, 1541-1575. [CrossRef]

15. Azar, A.T.; Koubaa, A.; Ali Mohamed, N.; Ibrahim, H.A.; Ibrahim, Z.F.; Kazim, M.; Ammar, A.; Benjdira, B.; Khamis, A.M.; Hameed, I.A.; et al. Drone Deep Reinforcement Learning; A Review. Electronics 2021, 10, 999. [CrossRef]

16. Sutton, R.; Barto, A. Reinforcement Learning: An Introduction. IEEE Trans. Neural Netw. 1998, 9, 1054-1054. [CrossRe

17. Littman, M.L. Markov games as a framework for multi-agent reinforcement learning. Mach. Learn. Proc. 1994, 157-163. [CrossRef

18. Gronauer, S.; Diepold, K. Multi-agent deep reinforcement learning: A survey. Artif. Intell. Rev. 2021, 55, 895-943. [CrossRef]

19. DrawExpress Lite [Gesture-recognition Diagram Application]. Available online: https://drawexpress.com/ (accessed on 27 February 2023).

20. Karur, K.; Sharma, N.; Dharmatti, C.; Siegel, J.E. A Survey of Path Planning Algorithms for Mobile Robots. Vehicles 2021, 3,448-468. [CrossRef]

21. Niu, Z.; Zhong, G.; Yu, H. A review on the attention mechanism of deep learning. Neurocomputing 2021, 452, 48-62. [CrossRef]

22. Shah, S.; Dey, D.; Lovett, C.; Kapoor, A. AirSim: High-Fidelity Visual and Physical Simulation for Autonomous Vehicles. arXiv 2017, arXiv:1705.05065.

23. Koenig, N.; Howard, A. Design and use paradigms for Gazebo, an open-source multi-robot simulator. In Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566), Sendai, Japan, 28 September-2 October 2004; Volume 3, pp. 2149-2154. [CrossRef]

24. Liu, C.H.; Chen, Z.; Zhan, Y. Energy-Efficient Distributed Mobile Crowd Sensing: A Deep Learning Approach. IEEE J. Sel. Areas Commun. 2019, 37, 1262-1276. [CrossRef]

25. Dai, Z.; Liu, C.H.; Han, R.; Wang, G.; Leung, K.; Tang, J. Delay-Sensitive Energy-Efficient UAV Crowdsensing by Deep Reinforcement Learning. IEEE Trans. Mob. Comput. 2021, 1233, 1-15. [CrossRef]

26. Wang, L.; Wang, K.; Pan, C.; Xu, W.; Aslam, N.; Hanzo, L. Multi-Agent Deep Reinforcement Learning-Based Trajectory Planning for Multi-UAV Assisted Mobile Edge Computing. IEEE Trans. Cogn. Commun. Netw. 2021, 7, 73-84. [CrossRef]

27. Liu, C.H.; Ma, X.; Gao, X.; Tang, J. Distributed Energy-Efficient Multi-UAV Navigation for Long-Term Communication Covera by Deep Reinforcement Learning. IEEE Trans. Mob. Comput. 2020, 19, 1274-1285. [CrossRef]

28. Bai, C.; Yan, P.; Yu, X.; Guo, J. Learning-based resilience guarantee for multi-UAV collaborative QoS management. Pattern Recognit. 2022, 122, 108166. [CrossRef]

29. Chen, Y.; Song, G.; Ye, Z.; Jiang, X. Scalable and Transferable Reinforcement Learning for Multi-Agent Mixed Cooperative-Competitive Environments Based on Hierarchical Graph Attention. Entropy 2022, 24, 563. [CrossRef] [PubMed]

30. Nemer, I.A.; Sheltami, T.R.; Belhaiza, S.; Mahmoud, A.S. Energy-Efficient UAV Movement Control for Fair Communication Coverage: A Deep Reinforcement Learning Approach. Sensors 2022, 22, 1919. [CrossRef] [PubMed]

31. Chen, D.; Qi, Q.; Zhuang, Z.; Wang, J.; Liao, J.; Han, Z. Mean Field Deep Reinforcement Learning for Fair and Efficient UAV Control. IEEE Internet Things J. 2021, 8, 813-828. [CrossRef]

32. Mou, Z.; Zhang, Y.; Gao, F.; Wang, H.; Zhang, T.; Han, Z. Three-Dimensional Area Coverage with UAV Swarm based on Deep Reinforcement Learning. IEEE Int. Conf. Commun. 2021, 39, 3160-3176. [CrossRef]

33. Liu, C.H.; Chen, Z.; Tang, J.; Xu, J.; Piao, C. Energy-efficient UAV control for effective and fair communication coverage: A deep reinforcement learning approach. IEEE J. Sel. Areas Commun. 2018, 36, 2059-2070. [CrossRef]

34. Li, S.; Jia, Y.; Yiang, F.; Qin, Q.; Gao, H.; Zhou, Y. Collaborative Decision-Making Method for Multi-UAV Based on Multiagent Reinforcement Learning. IEEE Access 2022, 10, 91385-91396. [CrossRef]

35. Ren, Z.; Zhang, D.; Tang, S.; Xiong, W.; heng Yang, S. Cooperative maneuver decision making for multi-UAV air combat based on incomplete information dynamic game. Def. Technol. 2022 . [CrossRef]

36. Wang, B.; Li, S.; Gao, X.; Xie, T. Weighted mean field reinforcement learning for large-scale UAV swarm confrontation. Appl. Intell. 2022, 1-16. [CrossRef]

37. Zhang, G.; Li, Y.; Xu, X.; Dai, H. Multiagent reinforcement learning for swarm confrontation environments. In Proceedings of the 12th International Conference, ICIRA 2019, Shenyang, China, 8-11 August 2019; Lecture Notes in Computer Science (including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Springer: Berlin/Heidelberg, Germany, 2019; Volume 11742 LNAI, pp. 533-543. [CrossRef]

38. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv, 2018, arXiv:1810.04805. [CrossRef]

39. Zhao, N.; Ye, Z.; Pei, Y.; Liang, Y.C.; Niyato, D. Multi-Agent Deep Reinforcement Learning for Task Offloading in UAV-Assisted Mobile Edge Computing. IEEE Trans. Wirel. Commun. 2022, 21, 6949-6960. [CrossRef]

40. Liu, Y.; Xie, S.; Zhang, Y. Cooperative Offloading and Resource Management for UAV-Enabled Mobile Edge Computing in Power IoT System. IEEE Trans. Veh. Technol. 2020, 69, 12229-12239. [CrossRef]

41. Cheng, Z.; Liwang, M.; Chen, N.; Huang, L.; Du, X.; Guizani, M. Deep reinforcement learning-based joint task and energy offloading in UAV-aided 6G intelligent edge networks. Comput. Commun. 2022, 192, 234-244. [CrossRef]

42. Sacco, A.; Esposito, F.; Marchetto, G.; Montuschi, P. Sustainable Task Offloading in UAV Networks via Multi-Agent Reinforcement Learning. IEEE Trans. Veh. Technol. 2021, 70, 5003-5015. [CrossRef]

43. Gao, A.; Wang, Q.; Liang, W.; Ding, Z. Game Combined Multi-Agent Reinforcement Learning Approach for UAV Assisted Offloading. IEEE Trans. Veh. Technol. 2021, 70, 12888-12901. [CrossRef]

44. Seid, A.M.; Lu, J.; Abishu, H.N.; Ayall, T.A. Blockchain-Enabled Task Offloading With Energy Harvesting in Multi-UAV-Assisted IoT Networks: A Multi-Agent DRL Approach. IEEE J. Sel. Areas Commun. 2022, 40, 3517-3532. [CrossRef]

Gao, A.; Wang, Q.; Chen, K.; Liang, W. Multi-UAV Assisted Offloading Optimization: A Game Combined Reinforcement Learning Approach. IEEE Commun. Lett. 2021, 25, 2629-2633. [CrossRef]

46. Qin, Z.; Liu, Z.; Han, G.; Lin, C.; Guo, L.; Xie, L. Distributed UAV-BSs Trajectory Optimization for User-Level Fair Communication Service With Multi-Agent Deep Reinforcement Learning. IEEE Trans. Veh. Technol. 2021, 70, 12290-12301. [CrossRef]

47. Xu, W.; Lei, H.; Shang, J. Joint topology construction and power adjustment for UAV networks: A deep reinforcement learning based approach. China Commun. 2021, 18, 265-283. [CrossRef]

48. Cheng, Z.; Liwang, M.; Chen, N.; Huang, L.; Guizani, N.; Du, X. Learning-based user association and dynamic resource allocation in multi-connectivity enabled unmanned aerial vehicle networks. Digit. Commun. Netw. 2022 . [CrossRef]

49. Zhu, Z.; Xie, N.; Zong, K.; Chen, L. Building a Connected Communication Network for UAV Clusters Using DE-MADDPG. Symmetry 2021, 13, 1537. [CrossRef]

50. Zhou, Y.; Ma, X.; Hu, S.; Zhou, D.; Cheng, N.; Lu, N. QoE-Driven Adaptive Deployment Strategy of Multi-UAV Networks Based on Hybrid Deep Reinforcement Learning. IEEE Internet Things J. 2022, 9, 5868-5881. [CrossRef]

51. Zhang, W.; Wang, Q.; Liu, X.; Liu, Y.; Chen, Y. Three-Dimension Trajectory Design for Multi-UAV Wireless Network With Deep Reinforcement Learning. IEEE Trans. Veh. Technol. 2021, 70, 600-612. [CrossRef]

52. Zhao, N.; Liu, Z.; Cheng, Y. Multi-Agent Deep Reinforcement Learning for Trajectory Design and Power Allocation in Multi-UAV Networks. IEEE Access 2020, 8, 139670-139679. [CrossRef]

53. Wu, F.; Zhang, H.; Wu, J.; Song, L. Cellular UAV-to-Device Communications: Trajectory Design and Mode Selection by Multi-Agent Deep Reinforcement Learning. IEEE Trans. Commun. 2020, 68, 4175-4189. [CrossRef]

54. Hu, J.; Zhang, H.; Song, L.; Schober, R.; Poor, H.V. Cooperative Internet of UAVs: Distributed Trajectory Design by Multi-Agent Deep Reinforcement Learning. IEEE Trans. Commun. 2020, 68, 6807-6821. [CrossRef]

55. Emami, Y.; Wei, B.; Li, K.; Ni, W.; Tovar, E. Joint Communication Scheduling and Velocity Control in Multi-UAV-Assisted Sensor Networks: A Deep Reinforcement Learning Approach. IEEE Trans. Veh. Technol. 2021, 70, 10986-10998. [CrossRef]

56. Wu, F.; Zhang, H.; Wu, J.; Han, Z.; Poor, H.V.; Song, L. UAV-to-Device Underlay Communications: Age of Information Minimization by Multi-Agent Deep Reinforcement Learning. IEEE Trans. Commun. 2021, 69, 4461-4475. [CrossRef]

57. Chen, B.; Liu, D.; Hanzo, L. Decentralized Trajectory and Power Control Based on Multi-Agent Deep Reinforcement Learning in UAV Networks. IEEE Int. Conf. Commun. 2022, 3983-3988. [CrossRef]

58. Wang, W.; Lin, Y. Trajectory Design and Bandwidth Assignment for UAVs-enabled Communication Network with Multi - Agent Deep Reinforcement Learning. In Proceedings of the 2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall), Norman, OK, USA, 27-30 September 2021; pp. 1-6. [CrossRef]

59. Zhou, Y.; Jin, Z.; Shi, H.; Wang, Z.; Lu, N.; Liu, F. UAV-Assisted Fair Communication for Mobile Networks: A Multi-Agent Deep Reinforcement Learning Approach. Remote Sens. 2022, 14, 5662. [CrossRef]

60. Zhang, Y.; Mou, Z.; Gao, F.; Jiang, J.; Ding, R.; Han, Z. UAV-Enabled Secure Communications by Multi-Agent Deep Reinforcement Learning. IEEE Trans. Veh. Technol. 2020, 69, 11599-11611. [CrossRef]

61. Katoch, S.; Chauhan, S.S.; Kumar, V. A review on genetic algorithm: Past, present, and future. Multimed. Tools Appl. 2021, 80, 8091-8126. [CrossRef] [PubMed]

62. Ma, X.; Hu, S.; Zhou, D.; Zhou, Y.; Lu, N. Adaptive Deployment of UAV-Aided Networks Based on Hybrid Deep Reinforcement Learning. In Proceedings of the 2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall), Victoria, BC, Canada, 18 November-16 December 2020; pp. 1-6. [CrossRef]

63. Wu, J.; Cheng, X.; Ma, X.; Li, W.; Zhou, Y. A Time-Efficient and Attention-Aware Deployment Strategy for UAV Networks Driven by Deep Reinforcement Learning. In Proceedings of the 2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall), Norman, OK, USA, 27-30 September 2021; pp. 1-5. [CrossRef]

64. Zeng, Y.; Zhang, R.; Lim, T.J. Wireless communications with unmanned aerial vehicles: opportunities and challenges. IEEE Commun. Mag. 2016, 54, 36-42. [CrossRef]

65. Hu, J.; Zhang, H.; Bian, K.; Song, L.; Han, Z. Distributed trajectory design for cooperative internet of UAVs using deep reinforcement learning. In Proceedings of the 2019 IEEE Global Communications Conference, GLOBECOM 2019-Proceedings, Waikoloa, HI, USA, 9-13 December 2019. [CrossRef]

66. Wu, F.; Zhang, H.; Wu, J.; Song, L.; Han, Z.; Poor, H.V. AoI Minimization for UAV-to-Device Underlay Communication by Multi-agent Deep Reinforcement Learning. In Proceedings of the GLOBECOM 2020-2020 IEEE Global Communications Conference, Taipei, Taiwan, 7-11 December 2020; pp. 1-6. [CrossRef]

67. Wu, F.; Zhang, H.; Wu, J.; Song, L. Trajectory Design for Overlay UAV-to-Device Communications by Deep Reinforcement Learning. In Proceedings of the 2019 IEEE Global Communications Conference (GLOBECOM), Waikoloa, HI, USA, 9-13 December 2019; pp. 1-6. [CrossRef]

68. Zhang, Y.; Zhuang, Z.; Gao, F.; Wang, J.; Han, Z. Multi-Agent Deep Reinforcement Learning for Secure UAV Communications. In Proceedings of the 2020 IEEE Wireless Communications and Networking Conference (WCNC), Seoul, Republic of Korea, 25-28 May 2020; pp. 1-5. [CrossRef]

69. Goh, K.C.; Ng, R.B.; Wong, Y.K.; Ho, N.J.; Chua, M.C. Aerial filming with synchronized drones using reinforcement learning Multimedia Tools and Applications. Multimed. Tools Appl. 2021, 80, 18125-18150. [CrossRef]

70. Du, W.; Guo, T.; Chen, J.; Li, B.; Zhu, G.; Cao, X. Cooperative pursuit of unauthorized UAVs in urban airspace via Multi-agent reinforcement learning. Transp. Res. Part Emerg. Technol. 2021, 128, 103122. [CrossRef]

71. ZHOU, W.; LI, J.; LIU, Z.; SHEN, L. Improving multi-target cooperative tracking guidance for UAV swarms using multi-agent reinforcement learning. Chin. J. Aeronaut. 2022, 35, 100-112. [CrossRef]

72. Yan, P.; Jia, T.; Bai, C. Searching and Tracking an Unknown Number of Targets: A Learning-Based Method Enhanced with Maps Merging. Sensors 2021, 21, 1076. [CrossRef]

73. Jiang, L.; Wei, R.; Wang, D. UAVs rounding up inspired by communication multi-agent depth deterministic policy gradient. Appl. Intell. 2022 . [CrossRef]

74. Wei, X.; Yang, L.; Cao, G.; Lu, T.; Wang, B. Recurrent MADDPG for Object Detection and Assignment in Combat Tasks. IEEE Access 2020, 8, 163334-163343. [CrossRef]

75. Zhou, W.; Liu, Z.; Li, J.; Xu, X.; Shen, L. Multi-target tracking for unmanned aerial vehicle swarms using deep reinforcement learning. Neurocomputing 2021,466, 285-297. [CrossRef]

76. Qie, H.; Shi, D.; Shen, T.; Xu, X.; Li, Y.; Wang, L. Joint Optimization of Multi-UAV Target Assignment and Path Planning Based on Multi-Agent Reinforcement Learning. IEEE Access 2019, 7, 146264-146272. [CrossRef]

77. Zhao, W.; Chu, H.; Miao, X.; Guo, L.; Shen, H.; Zhu, C.; Zhang, F.; Liang, D. Research on the multiagent joint proximal policy optimization algorithm controlling cooperative fixed-wing uav obstacle avoidance. Sensors 2020, 20, 4546. [CrossRef] [PubMed]

78. Lin, J.S.; Chiu, H.T.; Gau, R.H. Decentralized Planning-Assisted Deep Reinforcement Learning for Collision and Obstacle Avoidance in UAV Networks. In Proceedings of the 2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring), Helsinki, Finland, 25-28 April 2021; pp. 1-7. [CrossRef]

79. Shen, G.; Lei, L.; Li, Z.; Cai, S.; Zhang, L.; Cao, P.; Liu, X. Deep Reinforcement Learning for Flocking Motion of Multi-UAV Systems: Learn From a Digital Twin. IEEE Internet Things J. 2022, 9, 11141-11153. [CrossRef]

80. Wang, D.; Fan, T.; Han, T.; Pan, J. A Two-Stage Reinforcement Learning Approach for Multi-UAV Collision Avoidance under Imperfect Sensing. IEEE Robot. Autom. Lett. 2020, 5, 3098-3105. [CrossRef]

81. Venturini, F.; Mason, F.; Pase, F.; Chiariotti, F.; Testolin, A.; Zanella, A.; Zorzii, M. Distributed Reinforcement Learning for Flexible and Efficient UAV Swarm Control. IEEE Trans. Cogn. Commun. Netw. 2021, 7, 955-969. [CrossRef]

82. Walker, O.; Vanegas, F.; Gonzalez, F. A Framework for Multi-Agent UAV Exploration and Target-Finding in GPS-Denied and Partially Observable Environments. Sensors 2020, 20, 4739. [CrossRef] [PubMed]

83. Katsikopoulos, K.; Engelbrecht, S. Markov decision processes with delays and asynchronous cost collection. IEEE Trans. Autom. Control. 2003, 48, 568-574. [CrossRef]

84. Arjona-Medina, J.A.; Gillhofer, M.; Widrich, M.; Unterthiner, T.; Hochreiter, S. RUDDER: Return Decomposition for Delayed Rewards. Adv. Neural Inf. Process. Syst. 2019, 32 .

85. Kim, K. Multi-Agent Deep Q Network to Enhance the Reinforcement Learning for Delayed Reward System. Appl. Sci. 2022, 12, 3520. [CrossRef]

86. Agarwal, M.; Aggarwal, V. Blind Decision Making: Reinforcement Learning with Delayed Observations. Proc. Int. Conf. Autom. Plan. Sched. 2021, 31, 2-6. [CrossRef]

87. Chen, B.; Xu, M.; Li, L.; Zhao, D. Delay-aware model-based reinforcement learning for continuous control. Neurocomputing 2021, 450, 119-128. [CrossRef]

88. Patrizi, N.; Fragkos, G.; Tsiropoulou, E.E.; Papavassiliou, S. Contract-Theoretic Resource Control in Wireless Powered Communication Public Safety Systems. In Proceedings of the GLOBECOM 2020-2020 IEEE Global Communications Conference, Taipei, Taiwan, 7-11 December 2020; pp. 1-6. [CrossRef]

89. Zhang, Y.; Mou, Z.; Gao, F.; Xing, L.; Jiang, J.; Han, Z. Hierarchical Deep Reinforcement Learning for Backscattering Data Collection With Multiple UAVs. IEEE Internet Things J. 2021, 8, 3786-3800. [CrossRef]

90. Litmaps [Computer Software]. 2023. Available online: https://www.litmaps.com/spotlight-articles/litmaps-2023-redesign (accessed on 27 February 2023).

91. Lillicrap, T.P.; Hunt, J.J.; Pritzel, A.; Heess, N.; Erez, T.; Tassa, Y.; Silver, D.; Wierstra, D. Continuous control with deep reinforcement learning. In Proceedings of the 4th International Conference on Learning Representations, ICLR 2016-Conference Track Proceedings, San Juan, Puerto Rico, 2-4 May 2016.

92. Lowe, R.; Wu, Y.; Tamar, A.; Harb, J.; Abbeel, P.; Mordatch, I. Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments. Adv. Neural Inf. Process. Syst. 2017, 30, 6380-6391.

93. Zhang, S.; Zhang, H.; He, Q.; Bian, K.; Song, L. Joint Trajectory and Power Optimization for UAV Relay Networks. IEEE Commun. Lett. 2018, 22, 161-164. [CrossRef]

94. Zhang, H.; Song, L.; Han, Z.; Poor, H.V. Cooperation Techniques for a Cellular Internet of Unmanned Aerial Vehicles. IEEE Wirel. Commun. 2019, 26, 167-173. [CrossRef]

95. Hu, J.; Zhang, H.; Song, L. Reinforcement Learning for Decentralized Trajectory Design in Cellular UAV Networks with Sense-and-Send Protocol. IEEE Internet Things J. 2019, 6, 6177-6189. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

免责声明/出版商注释: 所有出版物中的声明、观点和数据仅代表个人作者或贡献者的观点，而不是MDPI 和/或编辑的观点。MDPI 和/或编辑不承担因内容中提及的任何想法、方法、指导或产品导致的人身或财产损害的责任。