基于模仿学习和强化学习的启发式多智能体路径规划

郭传友，刘志飞，田景志，刘先忠
（中国人民解放军 61150 部队，陕西　榆林　719000）

摘　要： 多智能体路径规划（Multi－Agent Path Finding，MAPF）扩展到大型动态环境中是一个越来越有挑战的问题。现实世界中，环境动态变化往往需要实时重新规划路径。在部分可观察环境中，使用强化学习方法学习分散的策略解决 MAPF 问题表现出较大潜力。针对智能体之间如何学会合作和环境奖励稀疏问题，提出基于模仿学习和强化学习的启发式多智能体路径规划算法。实验表明，该方法在高密度障碍环境中具有较好的性能和扩展性。

关键词： 多智能体路径规划；强化学习；模仿学习；启发式
中图分类号： TP181　　**文献标识码：** A　　**DOI：** 10.19358/j.issn.2097－1788.2024.09.006
引用格式： 郭传友，刘志飞，田景志，等. 基于模仿学习和强化学习的启发式多智能体路径规划［J］. 网络安全与数据治理，2024，43(9)：33－40.

Heuristic multi-agent path finding VIA imitation learning and reinforcement learning

Guo Chuanyou，Liu Zhifei，Tian Jingzhi，Liu Xianzhong
（Chinese People's Liberation Army 61150 Unit，Yulin 719000，China）

Abstract： The extension of multi-agent path finding（MAPF）to large-scale dynamic environment is an increasingly challenging problem. In the real world，dynamic changes in the environment often require real-time re planning. Using reinforcement learning method to learn decentralized strategies in some observable environments shows great potential to solve MAPF problems. A heuristic multi-agent path planning algorithm based on imitation learning and reinforcement learning is proposed to address the problems of how intelligent agents learn to cooperate and sparse environmental rewards. Experiments show that this method has good performance and scalability in high-density obstacle environment.

Key words： multi-agent path finding；reinforcement learning；imitation learning；heuristic

0　引言

MAPF 是对不同起始位置的多个智能体到他们各自目标位置的路径规划问题，关键约束是在保证智能体之间互相不碰撞的前提下到达目标位置，并保证路径规划的速度和质量。MAPF 在实际场景中有许多应用，如大型仓库管理[1－2]、数字游戏[3]、火车调度[4]、城市道路网络[5]、多机器人系统[6]等，更多实际应用可参考文献[7]。近年来，越来越多的团队对 MAPF 展开研究[8－11]，MAPF 取得了突破性进展，尤其是基于强化学习（Reinforcement Learning，RL）方法应用到 MAPF 问题中取得了较好效果，国内对 MAPF 问题的研究也越来越浓厚。

求解 MAPF 的最优解已经被证明是 NP-Hard 问题[12]。传统方法将 MAPF 规约为其他已解决的问题如 SAT[13]，或使用基于搜索的算法来解决，经典方法有增强的搜索[14]、基于冲突的搜索[15]以及改进的变体[16]等。然而，随着环境的动态变化和智能体数量的增加，搜索空间巨大对传统 MAPF 算法构成挑战。基于搜索的 MAPF 算法通过引入优先规划、大领域搜索和复杂的启发式函数来优化改进 MAPF 算法，前沿的算法有 EECBS[17]、CCBS[18]、MOA＊[19]、MAPF-ML-LNS[20]。这些算法能解决 3 000 多个智能体规模的 MAPF 问题，而且规划效率和质量较高，但这些集中式规划算法不能实时规划路径，可扩展性差。最近，分散式执行的强化学习方法应用于解决 MAPF 问题表现出较大的潜力，每个智能体根据局部观察分散执行策略。

RL 智能体在大型环境中和环境互动时，只有达到目标才可以获取奖励，而到达目标的过程中奖励稀疏，学

习效率不高，训练时间长，智能体还可能陷入死胡同。PRIMAL（Pathfinding via Reinforcement and Imitation Multi-Agent Learning）[21]采取集中式 MAPF 规划器生成专家演示路径，训练过程中结合了模仿学习和强化学习，加速了学习过程，但计算比较耗时，求解质量还需提高。G2RL（Globally Guided RL）[22]给予每个智能体额外的奖励遵循单智能体最短路径，但这可能会误导智能体，因为到达目标位置的路径不是唯一的，这会影响智能体和其他智能体之间的协调合作。DHC（Distributed Heuristic multi-agent path finding with Communication）[23]使用多条潜在路径作为智能体路径的启发式输入，并采用图卷积网络来加强智能体之间的通信，促进智能体之间的显式协调，但学习速度较慢。为了解决上述问题，本文提出了基于强化学习和模仿学习的启发式多智能体路径规划算法（Heuristic multi-agent path planning via Imitation and Reinforcement Learning，HIRL），在智能体的观察中加入额外的目标向量，并嵌入从目标源到智能体的多条潜在最短路径作为神经网络的输入，使用模仿学习来促进智能体之间的隐式协调，引入目标牵引的奖励函数来鼓励智能体进行有效的探索，当智能体向目标方向移动时给予正奖励。智能体依据自己的局部观察来做出决策，不需要学习联合动作值，因此具有很好的可扩展性。本文采用的主要方法如下：

（1）采用模仿学习框架加速智能体学习，促进智能体之间的隐式协调，而不需要智能体之间的显式通信。

（2）采用智能体到目标位置的方向向量作为智能体观察的额外信息。

（3）引入目标牵引的奖励函数，鼓励智能体朝着目标方向进行有效的探索。

（4）嵌入了从目标源到智能体多条最短路径作为神经网络的输入，能更有效地避免智能体之间的冲突和死锁情况发生。

（5）使用部分可观察的环境，智能体根据有限视野的观察决策行动，更加符合现实世界的环境。

1　相关工作

1.1　多智能体路径规划

关于 MAPF 有许多不同的定义和假设，以经典的 MAPF 为例，对 MAPF 进行阐述。

k 个智能体的经典 MAPF[24]被定义为一个元组（G，s，t）。其中 $G=(V, E)$ 是一个无向图，无向图中的节点 $v\in V$ 是智能体可以占据的位置，边 $(n, n')\in E$ 表示智能体从节点 n 移动到 n'的连线。k 代表问题中智能体的数量，即智能体 $\{a_1, a_2, \cdots, a_k\}$，$s$ 是初始位置的集合，每个智能体都有一个起始位置 $s_i\in s$，t 是目标位置的集合，每个智能体都有一个和目标位置 $t_i\in t$。

在经典 MAPF 中，时间被离散为时间步长。在每个时间步长中，每个智能体可以执行一个动作，一般有五种类型的动作：向上、向下、向左、向右和等待。

一个单智能体的路径规划是从起始位置到目标位置一系列动作的集合 $\pi=(a_1, a_2, \cdots, a_n)$，$k$ 个智能体的路径规划问题就是 k 条路径的集合 $\prod=\{\pi_1, \pi_2, \cdots, \pi_k\}$。其中第 i 个智能体对应路径 π_i。

MAPF 有很多解决方案，在许多实际应用中，需要一个有效的目标函数来优化 MAPF 问题。用来评估 MAPF 解决方案的最常见的三个目标函数是：

$$\max_{1\leqslant i\leqslant k} t(\pi_i) \tag{1}$$

$$\sum_{1\leqslant i\leqslant k} t(\pi_i) \tag{2}$$

$$\sum_{1\leqslant i\leqslant k} l(\pi_i) \tag{3}$$

其中目标函数（1）表示最晚到达目标位置的智能体所花费的时间，目标函数（2）表示所有智能体到达目标位置的时间总和，目标函数（3）表示所有智能体到达目标位置的路径长度总和。

1.2　多智能体强化学习

智能体通过强化学习，根据当前策略与环境进行交互，获取环境下一到达状态和该动作奖励，计算并更新策略，目标是最大化累积奖励。

1.2.1　单智能体强化学习

如果环境满足马尔可夫性质，如式（4）所示，RL 可以建模为一个马尔可夫决策过程（Markov Decision Process，MDP）。

$$P(s_{t+1}\mid s_t, s_{t-1}, s_{t-2}, \cdots, s_0)=P(s_{t+1}\mid s_t) \tag{4}$$

其中，s_t 表示时间步 t 时的状态，P 表示状态转移函数。

MDP 可以用式（5）来表示。

$$(S, A, R, \rho, \gamma) \tag{5}$$

其中，S 表示状态空间（$s_t\in S$）、A 表示动作空间，$a_t\in A$；R 表示奖励空间（$r_t\in R$）；$\boldsymbol{\rho}$ 表示状态转移矩阵（$\boldsymbol{\rho}_{SS'}=P[s_{t+1}=s'\mid s_t=s]$）；$\gamma$ 表示折扣因子，它用于表示及时奖励对未来奖励的影响程度。在 RL 中，有两个重要的概念：状态价值函数和动作价值函数。

状态价值函数：衡量智能体所处状态的好坏，用式（6）表示。

$$V_\pi(s)=\sum_{a\in A}\pi(a\mid s)[r(s,a)+\gamma\sum_{s'\in S}\rho(s'\mid s,a)V_\pi(s')] \tag{6}$$

其中，s'表示 s 下一时刻的状态，π 表示策略。

动作价值函数：衡量智能体采取特定动作的好坏，

用式（7）表示。

$$Q_\pi(s,a) = r(s,a) + \gamma \sum_{s' \in S} \rho(s' \mid s,a) \sum_{a \in A} \pi(a \mid s') Q_\pi(s',a) \tag{7}$$

RL 可分为两种基本方法：

（1）基于值方法（Value-based methods）：基于值的方法依赖于智能体所处的状态值，最优策略是最大值函数，用式（8）表示。

$$V^* (s) = \max_\pi V_\pi (S) \quad \forall s \in S \tag{8}$$

两种最主要的方法是值迭代和 Q-learning，状态转移概率存在时使用值迭代方法，状态转移概率未知时使用 Q-learning 方法。

值迭代试图最大化价值函数，用式（9）表示。

$$V_{k+1}(s) = \max_{a \in A} [r(s,a) + \gamma \sum_{s' \in S} \rho(s' \mid s,a) V_k(s')] \tag{9}$$

Q-learning 通过贝尔曼方程来更新动作值，用式（10）表示。

$$V_{k+1}(s, a) = Q_k(s, a) + \alpha [r_k + \gamma \max_a Q_k(s', a) - Q_k(s, a)] \tag{10}$$

其中，α 表示学习率。

（2）基于策略方法（Policy-based methods）：基于策略的方法不是计算值函数，而是直接搜索最优策略。最常用的基于策略的方法是 REINFORCE，该方法的模型是关于 θ（π_θ（$a \mid s$））的函数，通过梯度上升来更新参数时收益最大化，用式（11）表示。

$$\theta = \theta + \nabla_\theta \alpha \gamma^t G_t \ln(\pi_\theta(A_t \mid s_t)) \tag{11}$$

1.2.2 部分可观察的马尔可夫模型

一个完全合作的多智能体强化学习（Multi-Agent Reinforcement Learning，MARL）任务可以用分布式部分可观测马尔可夫决策过程（Dec-POMDP）[25] 来描述。Dec-POMDP 可由元组 $G = (n, S, U, P, r, Z, O, \gamma)$ 表示。其中 n 表示智能体的数量；$s \in S$ 表示状态；$u^a \in U$ 表示智能体的动作；$u^a \in U \equiv U^n$ 表示智能体的联合动作集合，$P(s' \mid s, u)$：$S \times U \times S \rightarrow [0, 1]$ 表示状态 s 下采取联合动作 u 转移到 s' 状态转移概率；$r(s, u)$：$S \times U \rightarrow R$ 表示奖励函数；$z \in Z$ 表示每个智能体的观察值由 $O(s, a)$：$S \times A \rightarrow Z$ 来描述；$\gamma \in (0, 1)$ 表示折扣因子。

1.2.3 模仿学习

模仿学习[26] 技术旨在模仿给定任务中的人类行为。在 IL 中，专家教学提供了有效信息，从而提高了学习效率，并且适用于复杂任务，而无需人工编程所需要的相关专业技能和知识。IL 在太难的学习挑战或者奖励稀疏问题中非常必要，IL 可以加速初始学习，或者有益于学习反复实验不容易学习的行为。IL 常用于指导状态空间的探索以及学习合作行为。下面介绍一种常用的从专家策略中学习的算法：行为克隆（BC）。

BC 方法是学习一个状态到动作的映射，而无需求解奖励函数。当这种映射是表示期望策略的最有效方式时，BC 方法是再现示教行为的有效方法。IL 的目标是在给定状态 x_i 和上下文 s 的情况下学习已给可以生成动作的策略。IL 方法是监督学习问题，策略可以通过一个简单的回归问题得到。IL 的目标是学习一个特定应用中的策略 π_θ。IL 的通用算法如算法 1 所示。

算法 1　行为克隆方法。

收集专家示教轨迹 D

选择策略表示 π_θ

选择目标函数 Γ

基于示教 D，调整策略参数 θ 优化 Γ

返回 优化的策略参数 θ

其中目标函数 Γ 表示专家示教和学徒策略的相似性，θ 为待调整的策略参数。

1.2.4 基于 MARL 的 MAPF

从单智能体 RL 到多智能体 RL，遇到的首个也是最重要的问题是维度灾难。因为状态空间的组合爆炸，需要大量的训练数据来收敛。最近许多工作都集中在分散的策略学习[27-28]，其中每个智能体学习自己的策略。第二，许多现有方法依赖于智能体之间的显式通信，在训练期间共享观察或者选定的动作[29-30]。强化学习方法应用到动态混合环境中往往由于任务时间长和环境奖励稀疏等原因性能下降，MAPPER（Multi-Agent Path Planning with Evolutionary Reinforcement learning in mixed dynamic environments）[31] 算法在中央规划器的指导下将长时间的任务分解为多个简单任务，以此来提高智能体在大规模环境中的性能。文献［32］将 MAPF 问题分解为两个子任务：到达目标和避免冲突。为了完成每一项任务，利用强化学习的方法，如深度蒙特卡洛树搜索、Q 混合网络和策略梯度方法，来设计智能体从观察值到动作的映射，最后将学习到的到达目标策略和避免碰撞策略混合为一个策略。MAGAT（Message-Aware Graph Attention Networks）[33] 提出一种消息感知图注意网络。该方法使用基于一键查询类似机制，该机制可以确定邻近智能体信息的相对重要性。MAGAT 接近集中式规划的专家性能，在不同智能体密度和不同通信带宽下都是非常有效的。上述方法设计较为复杂，实现难度较高。

2 多智能体路径规划问题的强化学习建模

2.1 环境设置

现实世界中许多应用如自动化仓库可以转换为部分

可观察的网格世界环境，每个智能体只观察到一定视野范围内（Field of View，FOV）的信息。考虑一个部分可观察的环境是面向真实场景的重要一步。使用离散的网格世界环境，在这个环境中，每个智能体具有部分可观察性。使用 FOV 可以允许智能体的策略扩展到更大规模的环境中，还可以减少神经网络的输入维度。将网格世界的可用信息分离到不同通道中，如图 1 所示。每个智能体的观察值分别由 FOV 内其它智能体位置、障碍物位置、自己位置、目标位置和目标向量组成。

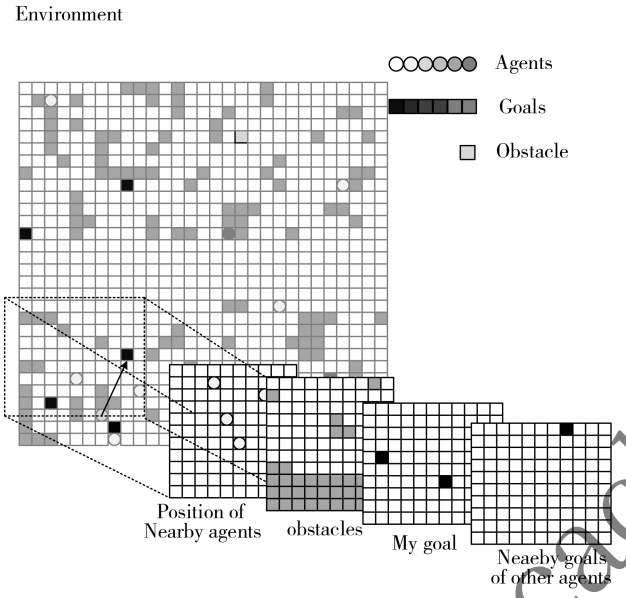


图 1　观察空间

2.2　动作空间

智能体在网格世界中采用离散的动作：上、下、左、右和等待共 5 个动作。有些动作可能是无效的，比如智能体碰到障碍或者走出边界，以及智能体之间发生位置冲突。在训练期间，屏蔽掉无效动作，仅从有效动作中采样，这种方法能够实现更快和更稳定的训练。

2.3　奖励函数

为了引导智能体尽快到达目标位置且智能体之间互相不碰撞，智能体在每个时间步都给予负奖励；当智能体之间发生位置冲突时，给予每个智能体负奖励；当智能体碰到障碍物时，给予负奖励；当智能体到达目标时，智能体获得较多正奖励。奖励函数如表 1 所示：

表 1　奖励函数

奖励类型	奖励
步数奖励（Step reward，SR）	-0.4
正向移动奖励（Positive movement reward，PR）	0.25
智能体碰撞奖励（Agent collision reward，AC）	-0.4
障碍物碰撞奖励（Obstacle collision reward，OC）	-0.02
到达目标奖励（Goal reached reward，GR）	1.0

智能体奖励用式（12）表示：

$$R_i = \sum \text{agentcollisions} \times \text{AC} + \sum \text{obstsclecollisions} \times \text{OC} + \text{GR} + \text{PR} + \text{SR} \tag{12}$$

2.4　网络结构

使用深度神经网络来逼近智能体的策略，将 FOV 映射到下一步采取的动作。本文采用 PRIMAL 算法中的网络结构，如图 2 所示。这个网络有多个输出，其中一个是实际的策略，另外两个用来训练。神经网络有三个输入：本地观察、目标向量和潜在选择路径，在神经网络连接之前进行预处理。表示局部观察值的四通道矩阵经过三个卷积层和最后一个池化层两个阶段，最后再经过三个卷积层，一个池化层。目标向量和潜在选择路径通过全连接层传递。这两个预处理输入串联通过两个全连接层最后输入到长短期记忆网络（LSTM）。输出层由 softmax 激活的策略神经元、值输入、和用于训练每个智能体的特征层组成。

3　方法

3.1　启发式潜在路径输入

本文引入 DHC 算法的四个启发式通道。和以往的给予最短路径奖励和离开路径惩罚不同，启发式通道从单个目标源路径中提取信息，而没有特殊的奖励设计。由于智能体的最短路径通常都不是唯一的，智能体根据多条最短路径的启发，自己学会最好的策略为最佳。四个通道对应上下左右四个动作，每个通道具有与 FOV 相同的大小，其中当智能体采取该通道的相关动作后更接近目标位置时，位置标记为 1。如图 3 所示。这四个通道和观察值一起作为模型的输入。

3.2　合作学习

协调学习分散策略的关键挑战是鼓励智能体学习无私的动作，因为智能体自私行为有时会造成阻塞的情况发生。特别是在有限的 FOV 和大规模智能体环境时，智能体之间的协调变得尤为重要。使用强化学习和模仿学习相结合的方法能有效解决这个问题。结合 RL 和 IL 可以导致更快更稳定的训练以及更高的求解质量。模仿学习的优势是帮助智能体快速识别智能体状态空间的有价值的区域，RL 的优势是可以通过探索这些区域来进一步改进策略。本文采用 PRIMAL 框架，如图 4 所示。在每一回合的开始以 0.2 的概率采用 IL，以 0.8 的概率采用 RL。

IL 的专家演示依赖集中式规划 ODrM* 算法[34]。ODrM* 算法使用基于搜索的启发式函数可以快速生成路径，但是求解质量还不高。通过专家演示路径得到每个智能体的观察和动作的轨迹，大多数智能体可以尽快地移动

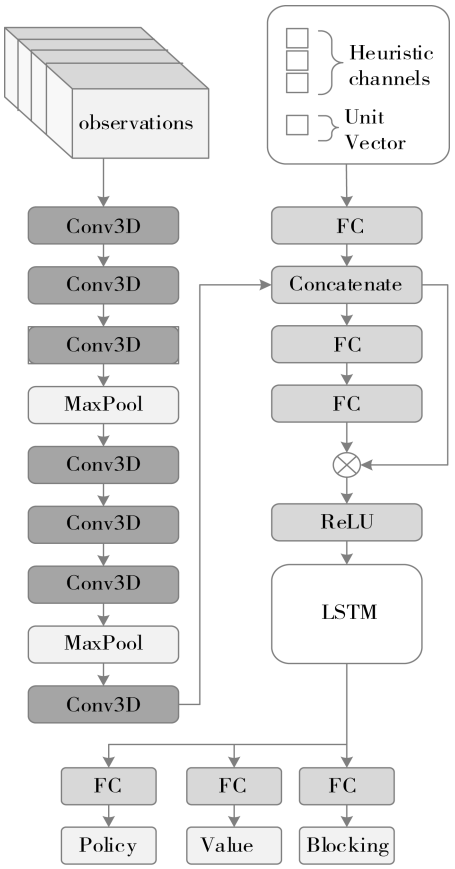


图 2　模型架构

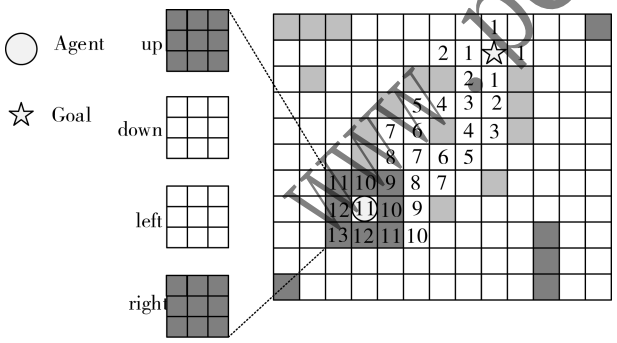


图 3　启发式通道

到目标位置。因此使用 IL 可以避免学习过程的振荡。如果没有 IL，学习速度会非常慢，并且有可能收敛到到达目标成功率很低的解决方案。

3.3　额外奖励目标方向移动的正向动作

智能体只有到达目标位置才有奖励，在巨大的搜索空间中，智能体奖励稀疏问题尤其突出。智能体不断优化自身策略来选择动作，来回移动到原来位置或者向目标的反方向移动都是不利于智能体策略更新的，在探索目标过程中无任何奖励。为了加快智能体的策略迭代，

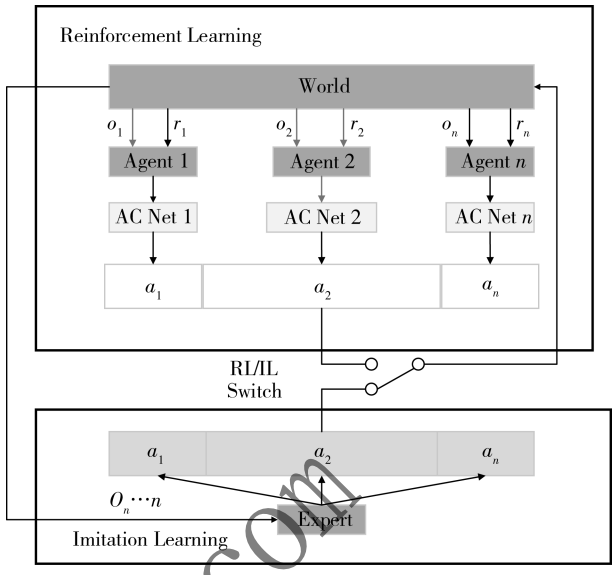


图 4　RL/IL 混合结构

给予智能体向目标方向移动的正奖励。为了防止智能体陷于目标方向的死胡同，奖励值不能太大，当智能体背向目标方向移动时有的时候是远离死胡同或者和其他智能体协调动作，所以背向目标方向的移动动作不给予奖励。将智能体动作方向与目标方向夹角小于 90°时定义为向目标方向移动，将智能体动作方向与目标方向夹角大于 90°时定义为背向目标方向移动。如图 5 所示，a_1 与目标方向向量夹角小于 90°，定义为正向移动，给予正奖励；a_2 与目标方向向量夹角大于 90°，定义为背向移动，无奖励。夹角公式如式（13）所示。

$$\angle\alpha = \arccos \frac{(x_{goal}-x_i)(x_{move}-x_i)+(y_{goal}-y_i)(y_{move}-y_i)}{[(x_{goal}-x_i)^2+(y_{goal}-y_i)^2]^{0.5}+[(x_{move}-x_i)^2+(y_{move}-y_i)^2]^{0.5}} \tag{13}$$

其中智能体坐标为（x_i，y_i），目标位置坐标为（x_{goal}，y_{goal}），智能体采取动作 a 后的下个位置坐标为（x_{move}，y_{move}）。

当智能体采取向上的动作时，（x_{move}，y_{move}）=（x_i，y_i+1）；

当智能体采取向下的动作时，（x_{move}，y_{move}）=（x_i，y_i-1）；

当智能体采取向左的动作时，（x_{move}，y_{move}）=（x_i-1，y_i）；

当智能体采取向右的动作时，（x_{move}，y_{move}）=（x_i+1，y_i）；

当智能体采取等待的动作时，（x_{move}，y_{move}）=（x_i，y_i）。

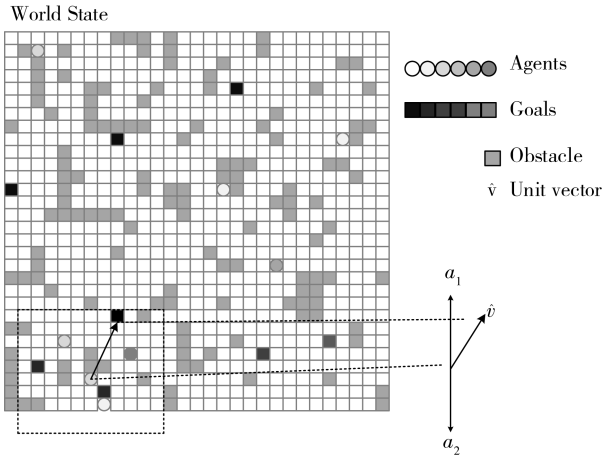


图 5　动作方向与目标方向夹角

4　实验

在 7 ×7 网格世界环境中进行实验，智能体数量为 4，障碍物密度为 0.3，对每轮采用 3 000 回合，对比分析 HIRL 与 PRIMAL、DHC 以及独立强化学习 IPPO 算法的性能。选择智能体到达目标的成功率、智能体之间碰撞次数和平均时间步长作为重要的衡量指标。当智能体到达目标成功率越高，表明算法性能越好。当智能体之间碰撞次数越少，表明智能体学到了成功的避开碰撞策略。当智能体到达目标的平均时间步长越小，表明策略越好。实验结果如图 6 ~ 图 8 所示。实验表明，HIRL 算法性能要明显优于现有算法。

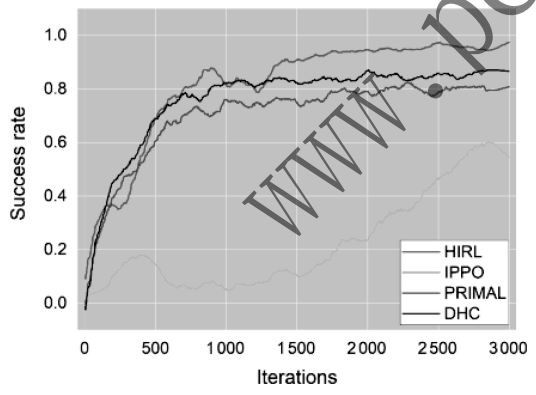


图 6　成功率

为了验证算法的扩展性，在更大规模的 40 ×40 的环境中进行实验，对 DHC 算法、PRIMAL 算法和本文的 HIRL 算法进行对比。智能体个数分别为 4 个、8 个、16 个、32 个，障碍物密度为 0.3。选择智能体到达目标的平均时间步长作为重要的衡量指标，平均时间步长越小表明策略越好。对每一轮采用 10 000 回合，实验表明，HIRL 算法的性能要优于 DHC 算法和 PRIMAL 算法，各个算法的平均时间步长见表 2。IL 在学习过程中能够很好地指导 RL 智能体。通过潜在多条可选择路径的启发，智能体可以更容易地找到较优路径，并且在更大智能体数量规模的环境中表现依然良好。

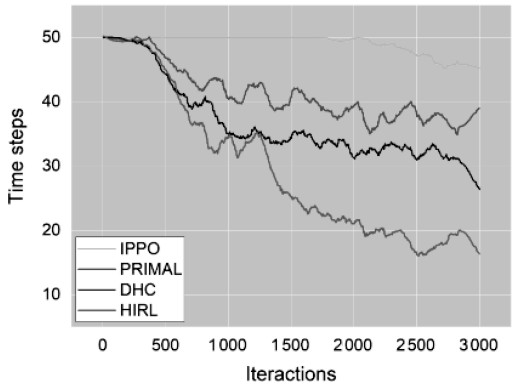


图 7　时间步长

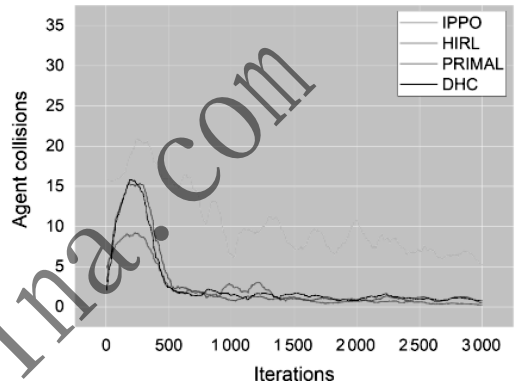


图 8　碰撞次数

表 2　平均时间步长

Agents	PRIMAL	DHC	HIRL
4	134	96	90
8	153	109	101
16	180	122	118
32	250	138	126

5　结论

本文提出基于模仿学习和强化学习的启发式多智能体路径算法，在经典的网格世界环境进行了测试比较。结合了模仿学习和强化学习，将多条潜在路径选择和目标向量作为启发式，让智能体自己学会策略。使用 IL 作为 RL 智能体的指导，智能体较快学习到高质量的策略，IL 也为智能体之间学会隐式协调，而不需要显式通信。通过目标向量，智能体能够在有限的 FOV 内使用额外有用的信息。使用潜在多条路径让智能体更好地学会协调合作，避免更多的死锁情况发生。引入目标牵引的奖励函数，能够让智能体更快地朝向目标方向进行探索。实验表明，该方法表现了较高的成功率和较快学会到达目

标策略。智能体之间如何通信和沟通协调在解决 MAPF 问题中显得尤为重要，未来研究工作将集中在加强智能体之间的通信时机、通信半径、通信对象和通信内容等。

参考文献

[1] YAKOVLEV K, ANDEYCHUK A. Any-angle pathfinding for multiple agents based on SIPP algorithm [C]//The Twenty-Seventh International Conference on Automated Planning and Scheduling (ICAPS 2017), 2017.

[2] LI J, TINKA A, KIESEL S, et al. Lifelong multi-agent path finding in large-scale warehouses [C]//AAMAS, 2020: 1898 – 1900.

[3] MA H, YANG J, COHEN L, et al. Feasibility study: moving non-homogeneous teams in congested video game environments [C]//Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, 2017, 13 (1): 270 – 272.

[4] 刘志飞，董强，赖俊，等. 多智能体强化学习在直升机机场调度中的应用 [J]. 计算机工程与应用，2023，59 (16)：285 – 294.

[5] CHOUDHURY S, SOLOVERY K, KOCHENDERFER M, et al. Coordinatedmulti-agent pathfinding for drones and trucks over road networks [J]. arXiv preprint arXiv: 2110.08802, 2021.

[6] CARTUCHO J, VENTURA R, VELOSOM. Robust object recognition through symbiotic deep learning in mobile robots [C]// 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2018: 2336 – 2341.

[7] FELNER A, STERN R, SHIMONY S E, et al. Search-based optimal solvers for the multi-agent pathfinding problem: summary and challenges [C]//International Symposium on Combinatorial Search, 2017, 8 (1).

[8] SURYNEK P, FELNER A, STERN R, et al. Anempirical comparison of the hardness of multi-agent path finding under the makespan and the sum of costs objectives [C]//Symposium on Combinatorial Search, 2016.

[9] BARTÁK R, SVANCARA J, ŠKOPKOVÁ V, et al. Multi-agent path finding on real robots: first experience with Ozobots [C]// Ibero-American Conference on Artificial Intelligence, 2018: 290 – 301.

[10] COHEN L, WAGNER G, CHAN D, et al. Rapid randomized restarts for multi-agent path finding solvers [C]//Eleventh Annual Symposium on Combinatorial Search, 2018.

[11] MA H, HARABOR D, STUCKEY P J, et al. Searching with consistent prioritization for multi-agent path finding [C]//Proceedings of the AAAI Conference on Artificial Intelligence, 2019, 33 (1): 7643 – 7650.

[12] YU J, LAVAALLE SM. Structure and intractability of optimal multi-robot path planning on graphs [C]//Twenty-Seventh AAAI Conference on Artificial Intelligence, 2013.

[13] SURYNEKP. Makespan optimal solving of cooperative path-finding via reductions to propositional satisfiability [J]. arXiv preprint arXiv: 1610.05452, 2016.

[14] STANDLEYT. Finding optimal solutions to cooperative path-finding problems [C]//Proceedings of the AAAI Conference on Artificial Intelligence, 2010, 24 (1): 173 – 178.

[15] SHARON G, STERN R, FELNER A, et al. Conflict-based search for optimal multi-agent pathfinding [J]. Artificial Intelligence, 2015, 219: 40 – 66.

[16] LI J, FELNER A, BOYARSKI E, et al. Improved heuristics for multi-Agent path finding with conflict-based search [C]// IJCAI. 2019, 2019: 442 – 449.

[17] LI J, RUML W, KOENING S. EECBS: abounded-suboptimal search for multi-agent path finding [C]//Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), 2021: 12353 – 12362.

[18] ANDREYCHUK A, YAKOVLEV K, SURYNEK P, et al. Multi-agent pathfinding with continuous time [J]. Artificial Intelligence, 2022: 103662.

[19] REN Z, RATHINAM S, LIKHACHEV M, et al. Enhanced-multi-objective A * using balanced binary search trees [J]. arXiv preprint arXiv: 2202.08992, 2022.

[20] HUANG T, LI J, KOENING S, et al. Anytimemulti-agent path finding via machine learning-guided large neigh-borhood search [C]//Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), 2022: 4127 – 4135.

[21] SARTORETTI G, KERR J, SHI Y, et al. PRIMAL: pathfinding via reinforcement and imitation multi-agent learning [J]. IEEE Robotics & Automation Letters, 2019, 4 (3): 2378 – 2385.

[22] WANG B, LIU Z, Li Q, et al. Mobile robot path planning in dynamic environments through globally guided reinforcement learning [J]. IEEE Robotics and Automation Letters, 2020, 5 (4): 6932 – 6939.

[23] MA Z, LUO Y, MA H. Distributed heuristic multi-agent path finding with communication [C]//2021 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2021: 8699 – 8705.

[24] STERN R, STURTEVANT N R, FELNER A, et al. Multi-agent pathfinding: definitions, variants, and bench-marks [C]// Twelfth Annual Symposium on Combinatorial Search, 2019.

[25] OLIEHOEK F A, SPAAN M T J, VLASSISN. Optimal and approximate Q-value functions for decentralized POMDPs [J]. Journal of Artificial Intelligence Re-search, 2008, 32: 289 – 353

[26] 李帅龙，张会文，周维佳. 模仿学习方法综述及其在机器人领域的应用 [J]. 计算机工程与应用，2019，55（4）：17－30.

[27] DE WITT C S，GUPTA T，MAKOVIICHUK D，et al. Is independent learning all you need in the starcraft multi-agent challenge [J]. arXiv preprint arXiv：2011. 09533，2020.

[28] ABED-ALGUNI B H，PAUL D J，CHALUP S K，et al. A comparison study of cooperative Q-learning algorithms for independent learners [J]. Int. J. Artif. Intell.，2016，14（1）：71－93.

[29] LOWE R，WU Y I，TAMAR A，et al. Multi-agent actor-critic for mixed cooperative-competitive environments [J]. Advances in Neural Information Processing Systems，2017，30：6382－6393.

[30] FOERSTER J，FARQUHAR G，AFOURAS T，et al. Counterfactual multi-agent policy gradients [C] //Proceedings of the AAAI Conference on Artificial Intelligence，2018，32（1）.

[31] LIU Z，CHEN B，ZHOU H，et al. Mapper：multi-agent path planning with evolutionary reinforcement learning in mixed dynamic environments [C]//2020 IEEE/RSJ International Conference on Intelligent Robots and Systems（IROS）. IEEE，2020：11748－11754.

[32] SKRYNNIK A，YAKOVLEVA A，DAVYDOV V，et al. Hybrid policy learning for multi-agent pathfinding [J]. IEEE Access，2021，9：126034－126047.

[33] LI Q，LIN W，LIU Z，et al. Message-aware graph attention networks for large-scale multi-robot path planning [J]. IEEE Robotics and Automation Letters，2021，6（3）：5533－5540.

[34] WAGNER G，CHOSET H. Subdimensional expansion for multirobot path planning [J]. Artificial Intelligence，2015，219（2）：1－24.

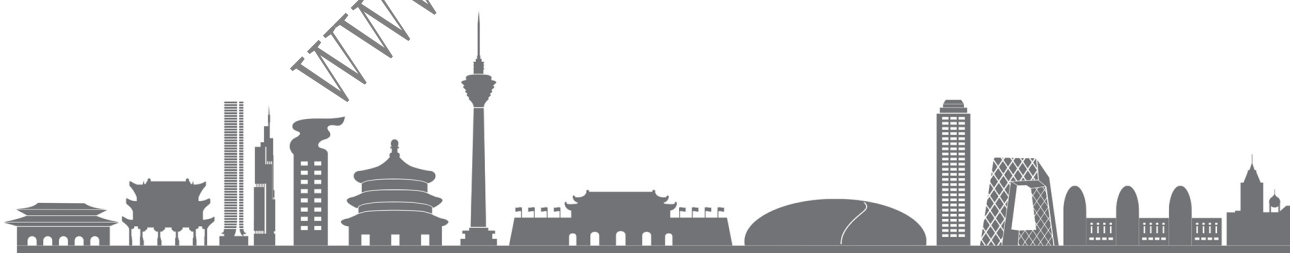
（收稿日期：2023－06－11）

作者简介：

郭传友（1972－），男，本科，高级工程师，主要研究方向：自动化技术、航空电子。

刘志飞（1985－），通信作者，男，硕士研究生，工程师，主要研究方向：智能化指挥控制。E-mail：1213281641@qq. com。

田景志（1986－），男，本科，工程师，主要研究方向：航空军械、自动化技术。



版权声明

凡《网络安全与数据治理》录用的文章，如作者没有关于汇编权、翻译权、印刷权及电子版的复制权、信息网络传播权与发行权等版权的特殊声明，即视作该文章署名作者同意将该文章的汇编权、翻译权、印刷权及电子版的复制权、信息网络传播权与发行权授予本刊，本刊有权授权本刊合作数据库、合作媒体等合作伙伴使用。同时，本刊支付的稿酬已包含上述使用的费用，特此声明。

《网络安全与数据治理》编辑部