多智能体路径规划综述

刘志飞[1],曹 雷[1],赖 俊[1],陈希亮[1],陈 英[2]
1.陆军工程大学 指挥控制工程学院,南京 210007
2.东部战区总医院 博士后科研工作站,南京 210007

摘 要:多智能体路径规划(multi-agent path finding,MAPF)是为多个智能体规划路径的问题,关键约束是多个智能体同时沿着规划路径行进而不会发生冲突。MAPF在物流、军事、安防等领域有着大量应用。对国内外关于MAPF的主要研究成果进行系统整理和分类,按照规划方式不同,MAPF算法分为集中式规划算法和分布式执行算法。集中式规划算法是最经典和最常用的MAPF算法,主要分为基于A*搜索、基于冲突搜索、基于代价增长树和基于规约四种算法。分布式执行算法是人工智能领域兴起的基于强化学习的MAPF算法,按照改进技术不同,分布式执行算法分为专家演示型、改进通信型和任务分解型三种算法。基于上述分类,比较MAPF各种算法的特点和适用性,分析现有算法的优点和不足,指出现有算法面临的挑战并对未来工作进行了展望。

关键词:多智能体路径规划;人工智能;搜索;分布式;强化学习
文献标志码:A **中图分类号:**TP181 doi:10.3778/j.issn.1002-8331.2203-0467

Overview of Multi-Agent Path Finding

LIU Zhifei[1], CAO Lei[1], LAI Jun[1], CHEN Xiliang[1], CHEN Ying[2]
1.College of Command and Control Engineering, Army Engineering University, Nanjing 210007, China
2.Postdoctoral Research Workstation of Eastern Theater Hospital, Nanjing 210007, China

Abstract: The multi-agent path finding (MAPF) problem is the fundamental problem of planning paths for multiple agents, where the key constraint is that the agents will be able to follow these paths concurrently without colliding with each other. MAPF is widely used in logistics, military, security and other fields. MAPF algorithm can be divided into the centralized planning algorithm and the distributed execution algorithm when the main research results of MAPF at home and abroad are systematically sorted and classified according to different planning methods. The centralized programming algorithm is not only the most classical but also the most commonly used MAPF algorithm. It is mainly divided into four algorithms based on A* search, conflict search, cost growth tree and protocol. The other part of MAPF which is the distributed execution algorithm is based on reinforcement learning. According to different improved techniques, the distributed execution algorithm can be divided into three types: the expert demonstration, the improved communication and the task decomposition. The challenges of existing algorithms are pointed out and the future work is forecasted based on the above classification by comparing the characteristics and applicability of MAPF algorithms and analyzing the advantages and disadvantages of existing algorithms.

Key words: multi-agent path finding; artificial intelligence; search; distributed; reinforcement learning

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。然而环境的动态变化和智

基金项目:国家自然科学基金(61806221)。

作者简介:刘志飞(1985—),男,硕士研究生,研究方向为多智能体强化学习和智能化指挥控制,E-mail:1213281641@qq.com;曹雷(1965—),男,博士,教授,博士生导师,研究方向为智能化指挥控制;赖俊(1979—),男,硕士,副教授,研究方向为智能化指挥控制;陈希亮(1985—),男,博士,副教授,研究方向为智能化指挥控制;陈英(1982—),女,博士,讲师,研究方向为模式识别和机器学习。

收稿日期:2022-03-24 **修回日期:**2022-07-04 **文章编号:**1002-8331(2022)20-0043-20

能体数量的增加,对传统MAPF算法是一个较大的挑战。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算法的性能和适用性。基于强化学习的MAPF算法在解决动态变化环境的MAPF表现出较大的潜力。国外关于MAPF的详细综述有文献[12-13],国内关于MAPF的详细综述有文献[14],但是这些综述只对经典MAPF算法进行归类整理和介绍,对近年来人工智能领域兴起的MAPF的算法没有系统地整理和分类。

按照MAPF规划方式不同,MAPF算法被分为集中式规划和分布式执行两种算法。集中式规划方法由一个中央控制器来为所有智能体规划路径,它的前提假设是中央规划器掌握了所有智能体的起始位置、目标位置和障碍物位置等信息。集中式规划算法是最经典和常用的MAPF算法,在求解的速度和质量上都达到较好的效果。分布式执行算法主要是基于强化学习的算法,前提假设是每个智能体只掌握了视野内(一定范围内)智能体和障碍物的位置等信息,智能体根据当前策略不断和环境进行交互,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,最后找到一个最大化累积奖励的动作序列,完成多智能体路径规划任务。这类算法可以扩展到高密度和动态的部分可观察的环境中,高效解决现实世界中的多智能体路径实时再规划问题。

MAPF的研究主要有两大方向,一是如何改进现有的算法,二是在实际应用中如何处理约束。在实际应用场景中要考虑机器的速度、加速度、转角,以及各种干扰的约束,而多智能体路径规划将这些设定进行抽象化,将运动控制离散为时间步,将研究的重点集中在求解速度和质量上。

本文首先对MAPF问题进行了阐述,概述了经典的集中式规划算法,详细分析了经典算法的原理,然后概述了深度强化学习,解析了主流的强化学习算法原理,将MAPF问题描述为强化学习问题,介绍了基于强化学习的MAPF算法研究进展。在此基础上,指出现有算法面临的挑战,指出了下一步要解决的问题和研究方向。

1　经典多智能体路径规划问题

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

1.1　定义

首先描述经典的MAPF问题。k个智能体的经典MAPF问题[12]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问题中,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$\Pi=\{\pi_1,\pi_2,\cdots,\pi_k\}$。其中第$i$个智能体对应路径$\pi_i$。

1.2　冲突类型

MAPF的首要目标是找到所有智能体的路径规划,且不存在冲突,为此引入了冲突的概念,常见冲突类型有4种,即图1所示[12]。

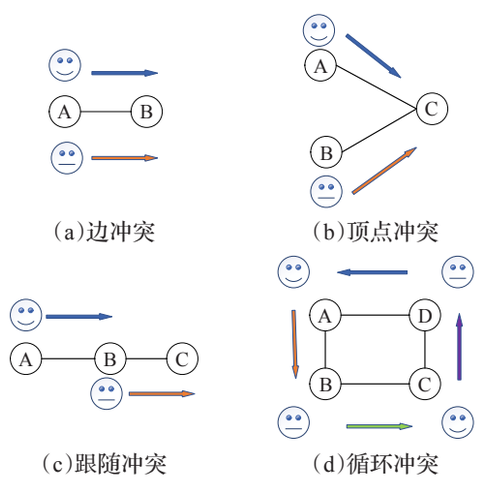

图1　冲突类型

Fig.1　Illustration of conflicts

1.3　目标函数

MAPF问题有很多解决方案,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。

2　集中式规划算法

MAPF集中式规划算法的前提假设是中央规划器掌握了全局信息,即所有智能体的起始位置、目标位置和障碍物位置信息等。MAPF集中式规划算法可分为

基于 A* 搜索、基于冲突搜索算法、代价增长树搜索算法和基于规约四种算法。集中式规划算法优缺点分析如表1所列。

2.1 基于 A* 搜索算法

2.1.1 A* 搜索算法

A* 搜索算法[15]是启发式搜索算法的代表,它是Dijikstra算法的扩展形式。A* 算法通常用来解决最短路径问题,相关工作总结于表2。

为了完整起见,先简要介绍背景。A* 在小规模智能体环境中是最佳搜索算法。它维护两个顶点列表:开放列表和关闭列表。最初将开始节点放入开放列表中,在每次迭代中,在开放列表中搜索相邻节点,并把起始节点放入关闭列表中。对于每个新生成的开放列表中的节点,A* 算法计算以下几个值:

(1) $g(n)$ 是从源节点到 n 节点的最短路径代价值。

(2) $parent(n)$ 是该节点的前一个节点。

(3) $h(n)$ 是 n 节点到目标节点的启发式路径代价估计值。

假设 $h^*(n)$ 是 n 节点到目标节点的完美启发式估计,如果已知每个节点 $h^*(n)$,就可以选择从源节点到目标节点的最短路径。A* 算法选择扩展开放列表里具有最小 $g(n)+h(n)$ 节点。

2.1.2 用 A* 搜索解决MAPF的挑战

A* 搜索在MAPF中的状态一般被称为 k-agent 状态空间,其状态数等于将 v 个智能体分别放置在 v 个不同节点状态数,一种放置方法对应于一种状态。一种最简单的方法将 A* 搜索启发式函数应用到MAPF中,是全局启发式函数等于所有智能体各自启发式函数之和。在 k-agent 搜索空间,最坏的情况下,搜索空间大小为 $|V|^k$,分支因子为 $\left(|E|/|V|\right)^k$,以一个20个智能体的200×200单元栅格四联通MAPF为例,搜索空间大小为 $40\,000^{20}$,分支因子为 4^{20}。对于 A* 算法来说,分支因子尤其是问题,因为 A* 必须沿着最优路径扩展所有的节点,因此 A* 算法在解决大规模智能体的MAPF问题时效率和质量都不高。

2.1.3 A* 搜索的扩展

Standley[16]对 A* 提出了两个非常有效的扩展来解决MAPF问题:因子分解和独立检测。

因子分解:第一个扩展称为算子分解(operator decomposition,OD)。OD设计用于改进 k-agent 搜索空间的分支因子指数增长的缺陷。在OD中,智能体按照任意的顺序进行排序。当展开源顶点 $\left(s(1),s(2),\cdots,s(k)\right)$ 时,只考虑一个智能体的行为,这将生成一组顶点,这些顶点表示时间步1中第一个智能体的可能位置,以及时间步0中所有其他智能体所占据的位置。这些顶点被添加到open列表中,当展开其中一个顶点时,只考虑第二个智能体的动作,生成一组新的顶点。这些顶点表示时间步1中第一个和第二个智能体的可能位置,其他智能体处在时间步0的位置。搜索工作继续以这种方式进行,直到搜索树上第 k 层状态节点能够表示所有智能体在时间步1的所有可能位置分布。第 k 层状态节点表示同一时间步长的所有智能体位置的节点称为full节

表1 集中式规划算法对比

Table 1 Comparison of centralized planning algorithms

算法	优点	缺点
基于 A* 搜索算法	原理简单,易于实现	空间代价高,求解速度慢,只适用于小规模智能体环境
基于冲突搜索算法	求解速度快,适合高密度和大规模智能体环境	实现难度高
代价增长树搜索算法	实现简单,速度快	在高密度智能体环境中易失效
规约算法	求解速度快	规约证明困难

表2 基于 A* 搜索算法

Table 2 Algorithms based on A* search

算法	主要机制	特点
A*	在Dijikstra算法基础上加入启发式函数	适用2~30个智能体规模环境,运行速度慢,时间代价和空间代价高
OD A*	通过每次只扩展单个智能体并引入中间节点来改进缺陷	加深了 A* 的搜索深度
ID A*	通过计算关联性,将问题分解为子问题进行求解以改进状态空间缺陷	大幅提高 A* 算法的执行效率
EPE A*	只扩展最优节点,当前节点修改启发式函数并加入开放列表	能够使用较小的计算成本,实现较优的结果
M*	动态搜索空间的分支因子	
OD rM*	同时改进了状态空间的指数增长和分支因子增长缺陷	适用于30~60个智能体规模环境
LM*	在规划单个智能体同时关注地图结构和可能发生冲突的其他智能体,然后通过将这种基于学习的单智能体规划器与 M* 集成	运行速度更快,成功率更高
MO A*	在搜索过程中的每个节点处维护一个边界集,以跟踪到达该节点的非支配、部分路径。通过在MO A* 搜索框架内逐步构建平衡二叉搜索树,有效维护多个目标的边界	运行速度比现有技术快一个数量级
LSS	松散同步搜索方法	如果存在最优解,则可以求得最优解

点,而其他所有顶点称为中间节点。继续搜索,直到到达表示目标 $(t(1),t(2),\cdots,t(k))$ 的 full 节点。有 OD 的 A^* 与无 OD 的 A^* 相比,其明显优势在于分支因子。OD 大幅缩减了 A^* 搜索的分支因子,OD A^* 引入了大量的中间状态节点,使得 A^* 搜索的深度加深了 k 倍。在 MAPF 中,由于加入启发式函数对中间状态节点进行剪枝,因此 OD 的引入是有益的。

独立检测:Standley 提出的第二个 A^* 扩展称为独立检测(independence detection,ID)。ID 试图将带有 k-agent 的 MAPF 问题分解为含有更少智能体的更小的 MAPF 问题。它的工作原理如下:首先,每个智能体为自己找到一个最优路径方案,同时忽略所有其他智能体,然后开始独立检测,如果一对智能体的计划之间存在冲突,这些智能体就合并为一个单一的元智能体。然后,用 A^*+OD 求出其中两个智能体的最优解,忽略其他所有智能体。这个过程以迭代的方式继续,在每次迭代中检测到单个冲突,合并冲突(元)智能体,然后用 A^*+OD 进行最优解决。当智能体的计划之间没有冲突时,进程停止。在最坏的情况下,ID 最终会将所有智能体合并到一个元智能体,并解决生成智能体的 MAPF 问题。ID 是一个非常通用的框架的 MAPF 求解器,ID 能够提高多数 MAPF 问题求解的效率。

M^* 算法[17]也像 A^* 一样搜索 k-agent 搜索空间。为了改进分支因子,M^* 动态地改变搜索空间的分支因子。最初,每当扩展一个节点时,它只生成一个节点,该节点对应于所有单个智能体最优路径,这将在 k 个智能体搜索空间中生成 k 条路径。由于智能体沿着各自的最优路径移动,可能会生成一个节点来表示一对智能体 i 和 j 之间的冲突。如果发生这种情况,智能体 i 和 j 将会加入到冲突集合中,然后重新展开搜索。在重启搜索时,智能体 i 和 j 会执行朴素的 A^* 搜索。一般情况下,M^* 中的一个节点存储冲突集,冲突集是一组智能体,它将为这些智能体生成所有动作组合。对于不在冲突集中的智能体,M^* 只考虑单个智能体其最优路径上的动作。递归 M^*(rM^*)是 M^* 的一个显著改进版本。rM^* 试图将冲突的智能体分割为没有冲突的智能体集,递归的解决生成的子问题。M^* 与 OD 相似,它限制了某些节点的分支因子。rM^* 与 ID 也有一些相似之处,因为它试图识别哪些智能体可以单独求解。然而,M^*、OD 和 ID 可以一起使用,rM^* 可以通过 ID 来寻找冲突元智能体的最优解,rM^* 可以用带有 OD 的 A^* 来搜索 k-agent 搜索空间,而不是简单的 A^*,后者被称为 OD rM^*。

现有基于搜索的方法,大部分工作都集中在如何处理智能体之间的冲突,文献[18]采取不同的方法,通过改进每个智能体的个人计划来减少冲突的数量,从而提高整体搜索效率。它在规划单个智能体同时关注地图结构和可能发生冲突的其他智能体,然后通过将这种基于学习的单智能体规划器与 M^* 集成,开发了一种称为 LM^* 的新型多智能体规划器。LM^* 需要解决的冲突更少,运行速度更快,成功率更高。文献[19]针对多目标最短路径问题,开发了 MOA^* 算法,该方法在搜索过程的每个节点处维护一个边界集,以跟踪到达该节点的非支配、部分路径。通过在 MOA^* 搜索框架内逐步构建平衡二叉搜索树,有效维护多个目标的边界,该方法比现有技术运行速度快一个数量级。文献[20]针对目前几乎所有基于 A^* 方法都假设每个智能体同时执行一个动作的问题,提出一种松散同步搜索的方法(LSS),LSS 扩展可基于 A^* 的 MAPF 以处理异步操作,其中智能体不一定同时启动和停止。LSS 是完整的,并且如果存在最优解,则可以找到最优解。$EPEA^*$(enhanced partial expansion)[21-22]对分支因子指数增长进行改进,与 A^* 不同之处在于 $EPEA^*$ 在扩展一个状态节点时只将一部分的后继状态节点加入 open 列表中。

2.2 基于冲突搜索算法

基于冲突搜索算法是目前最常用的算法,目前求解 MAPF 速度和质量最好的算法大都是在基于冲突搜索算法上进行改进和优化,相关工作总结于表 3。

2.2.1 经典基于冲突搜索算法

基于冲突搜索方法(CBS)[23]是解决 MAPF 最热门的方法之一,它包含高层搜索和低层搜索,它构建一棵二叉搜索树查找解。在根节点为所有智能体单独规划路径,然后通过添加限制的方式消解冲突,每个节点规划路径考虑节点被添加的限制并忽略其他智能体。它的特点在于求解速度快,相比其他方法能够求解更大规模的问题,缺点是实现难度较高。原有的用 A^* 整体规划求解 MAPF 问题需要在扩展同时考虑各个智能体之间的冲突,生成大量无意义的节点,影响搜索的效率。CBS 通过添加限制解决冲突,求解速度更快。

2.2.2 增强的基于冲突搜索算法

增强的基于冲突搜索算法(ICBS)[24]在 CBS 的基础上总结和提出了优化为 Meta-Agent 和规避冲突,此外 ICBS 进一步提出合并后重新搜索和有限消解关键冲突两种优化方法。Meta-Agent 主要思想是将冲突的两个智能体合并成一个 Meta-Agent 来重新规划以消解冲突,代替了通过分裂操作为两个智能体添加新约束的方法。合并后的 Meta-Agent 被当作一个复合的智能体,由于其他智能体都没有发生变化,只需要为新生成的 Meta-Agent 重新规划路径。ICBS 进一步减少了现有基于 CBS 方法的运行时间。

2.2.3 基于冲突搜索算法的变体

CBS 是优化多智能体路径规划的有效方法。然而,在具有许多智能体的高度竞争图中,CBS 的性能会迅速

表3 基于冲突搜索算法

Table 3 Conflict based search algorithms

算法	主要机制	特点
CBS	它包含高层搜索和低层搜索,它构建一棵二叉搜索树查找解。在根节点为所有智能体单独规划路径,然后通过添加限制的方式消解冲突,每个节点规划路径考虑节点被添加的限制并忽略其他智能体	求解速度快,适用于120~200个智能体规模环境
ICBS	在CBS的基础上总结和提出了优化为Meta-Agent和规避冲突,此外ICBS进一步提出合并后重新搜索和有限消解关键冲突两种优化方法	进一步减少CBS的运行时间
Lazy-CBS	使用惰性构造的约束规划模型替换CBS的高级求解器,使用核心引导的深度优先搜索太探索冲突空间,并沿着每个分支检测可重复使用的分支	显著改进成本度量下的最优MAPF问题
HCBS	为高层搜索增加了一个启发式函数,以更好地对约束树进行剪枝	能够使用较小的计算成本,实现较优的结果
ECBS	放宽CBS运行时的最优解决方案条件	运行时间大幅减少,同时解决方案质量微小损失
FECBS	通过在低层搜索中使用更宽松的次优边界,进一步减少需要在高层解决的冲突数量,同时仍提供有界次优解决方案	FECBS可以在5 min内解决比ECBS更多的MAPF实例
ASB-ECBS	一种自适应智能体特定的次优边界方法(ASB-ECBS),可以静态或者动态执行,可以根据单个智能体的要求分配次优界限	显著改善了运行时间,同时减少了搜索空间
EECBS	对ECBS进行了改进,它包含了显示估计搜索和三个启发式函数。第一个启发式函数是 A^* 中的 $h(n)$,第二个启发式函数计算冲突的个数,第三个是启发式函数是在线学习方法,在搜索过程中观察前面的 $h(n)$ 和冲突个数的误差,然后反馈矫正	可以解决1 000个智能体的MAPF实例,而且接的质量与最优解只有2%的误差
ML-guide CBS	由一个用于冲突选择的预言机做出决策,并学习一种由线性排序函数表示的冲突选择策略,该函数可以准确快速地模仿预言机的决策	显著提高了CBS求解器的成功率、搜索树大小和运行时间
CCBS	一种非连续时间、非网格域和非离散时间步长假设下的基于冲突的搜索算法	考虑了智能体的几何形状和连续时间,适用性更广,但求解速度慢
IDCBS	在不耗尽内存且没有严格时间限制情况下执行	在处理CBS节点时使用了增量方法,IDCBS比CBS要快得多

下降。发生这种情况的原因之一是CBS没有检测到独立的子问题。文献[25]提出惰性CBS(Lazy-CBS),这是一种新的MAPF方法,它使用惰性构造的约束规划模型替换了CBS的高级求解器。它使用核心引导的深度优先搜索来探索冲突空间,并沿着每个分支检测可重复使用的分支,这有助于快速确定可行的解决方案。Lazy-CBS可以显著改进成本度量下的最优MAPF问题。文献[26]引入了一种扩展的冲突分类,首先解决子节点成本值大于待拆分节点成本值的冲突,并提出一种识别这种冲突的方法,通过使用解决某些冲突的成本的信息来增强所有已知的CBS启发式算法,而这些信息值需要很小的计算开销,扩展的冲突分类和改进的启发式方法都是CBS更加高效。HCBS[27]为高层搜索增加了一个启发式函数,以更好地对约束树进行剪枝。增强型基于冲突搜索(ECBS)[28]是近似最优MAPF算法,它在低层搜索和高层搜索中引入次优性,在CBS的低层搜索中可以是任何最优路径算法,比如 A^*。文献[29]提出灵活的ECBS(FECBS),通过在低层搜索中使用更宽松的次优边界,进一步减少需要在高层解决的冲突数量,同时仍提供有界次优解决方案。FECBS不要求每条路径的成本是有界次优的,而是要求路径的总成本是有界次优的,因此可以根据需要在不同智能体之间自由分配成本。FECBS可以在5 min内解决比ECBS更多的MAPF实例。ECBS算法将解决方案质量折中到一个常数因子,以获得显著的运行时间的改进,但是ECBS对所有的智能体都使用固定的全局次优界限,而不管他们的偏好如何。实际上,随着智能体的增加,运行时性能会下降。文献[30]针对此问题提出了一种自适应智能体特定的次优边界方法(ASB-ECBS),可以静态或者动态执行,可以根据单个智能体的要求分配次优界限。ASB-ECBS显著改善了运行时间,同时减少了搜索空间。EECBS[31]算法对ECBS进行了改进,它包含了显示估计搜索和三个启发式函数。第一个启发式函数是 A^* 中的 $h(n)$,第二个启发式函数计算冲突的个数,第三个是启发式函数是在线学习方法,在搜索过程中观察前面的 $h(n)$ 和冲突个数的误差,然后反馈矫正。此外EECBS还增加了近年来一些新的CBS改进技术。通过实验对比分析,CBS能解决智能体的个数小于200,而EECBS可以达到1 000个以上,而且解的质量与最优解只有2%的误差。文献[32]提出一种用于选择冲突选择的机器学习框架:ML-guide CBS。该方法由一个用于冲突选择的预言机做出决策,并学习一种由线性排序函数表示的冲突选择策略,该函数可以准确快速地模仿预言机的决策。ML-guide CBS算法显著提高了CBS求解器的成功率、搜索树大小和运行时间。文献[33]提出了一种非连续时间、非网格域和非离散时间步长假设下的基于冲

突的搜索算法(CCBS),CCBS是一种可以处理非单元动作持续时间、连续时间、非网格域和具有集合形状的智能体的MAPF算法。由于CCBS考虑了智能体的几何形状和连续时间,它可能比基于网格的解决方案慢,但求解结果仍然是最优和完整的。基于冲突的搜索CBS变体通常使用某种形式的 A^* 搜索类计算MAPF解决方案。然而,这种方法经常受到严格的时间限制,以避免耗尽可用内存。文献[34]提出一种CBS的迭代延伸变体(IDCBS),它可以在不耗尽内存且没有严格时间限制情况下执行,由于在处理CBS节点时使用了增量方法,IDCBS比CBS要快得多。

2.3 代价增长树搜索算法

代价增长树搜索(increasing cost tree search,ICTS)[35]算法将MAPF问题分解为两个问题:找到每个智能体的代价和找到这些代价的有效解决问题方案,这部分工作总结于表4。

ICTS不直接搜索 k-agent 搜索空间。它将两个搜索过程结合在一起:高层搜索和低层搜索。高层搜索的目的是找出所有智能体的单个最优路大小,低层搜索目的是对高层搜索的状态节点 $(c_1,c_2,\cdots,c_k)$ 进行验证是否存在一组最优解。低层搜索首先计算每个智能体从起始位置到目标位置花费代价 c_i ,使用MDD存储。所有MDD的交叉乘积结果就是所有满足状态节点 $(c_1,c_2,\cdots,c_k)$ 的路径集合。MDD交叉乘积是 k-agent 搜索空间的一个子空间。ICTS搜索MDD交叉乘积以获得有效的解决方案。由于搜索方法是解决满足问题而不是优化问题,所以通常采用简单的深度优先搜索来实现。扩展的增加成本搜索(E-ICTS)[36]是ICTS的扩展。E-ICTS是一种两级算法,它在高层搜索递增成本树,在低层搜索 k 个MDD的所有时间重叠的动作,以找到一组 k 个不冲突路径。E-ICTS在具有离散运动模型中可以在短时间内实现比ICTS更高质量的次优解。ICTS在处理连续时间域内的对称冲突仍然难以解决,文献[37]引入了一种新的算法,即基于冲突的增加成本树搜索(CBICS),该算法结合了CBS和ICTS的优势,在单位时间和连续时间域中,CBICS通常比CBS和ICTS更快地找到解决方案。

2.4 基于规约算法

规约方法是从理论到实践的各种计算机领域中使用的最突出的技术之一。MAPF问题是NP-hard的问题[38],可以将MAPF问题规约为其他已解决的标准问题,如布尔可满足性(satisfaction fiability,SAT)、约束满足问题(constraint satisfaction problems,CSP)、约束优化问题(constraint optimization problems,COP)、答案集编程(answer set programming,ASP)。许多MAPF问题可建模为CSP或者COP。使用规约算法的主要好处是当前通用的规约求解器非常高效,并且变得越来越好。特别是现在的SAT求解器非常高效,能处理上万个变量。Surynek[39]探索了使用五种不同的SAT建模MAPF的方法,展示了不同的建模方法对SAT求解器运行时间的影响。文献[40]使用一种更高级的规约方法Picat[41]建模了集中MAPF的变体。一种用Picat编写的规约方法可以用SAT自动求解。文献[42]将MAPF问题规约为ASP问题,文献[43]将MAPF问题规约为SAT module theory(SMT),文献[44]将MAPF问题规约为多物品网络流问题。

2.5 其他集中式规划算法

其他集中式规划算法主要是两种或者两种以上算法的相融合算法。特别是将基于搜索的算法和机器学习相结合,使算法在性能上有很大提升,相关工作总结于表5。

文献[45]首先快速找到初始解决方案,然后通过大领域搜索(LNS)反复重新规划智能体子集。它通过随机破坏启发式生成重新规划的智能体子集,但并非所有的智能体子集都能显著提高解决方案的质量。文献[46]在MAPF-LNS算法基础加以改进,使用机器学习(ML)来学习如何从子集集合中选择智能体子集,以便找到更高质量的解决方案。MAPF-ML-LNS在改进解决方案的速度和质量方面均优于MAPF-LNS。文献[47]提出一种基于大领域搜索的新算法(MAPF-LNS2),从一组包含冲突的路径开始,MAPF-LNS2重复选择一个冲突智能体子集并重新规划他们的路径以减少冲突的数量,直到路径变得无冲突。MAPF-LNS2与最先进的随机重启的优先规划和EECBS相比,运行速度明显较快。MAPF-LNS2在大多数MAPF实例中,运行时间限制为5 min。文献[48]提出协作时间路线图方法(CTRMs),CTRMs使每个智能体能够以考虑其他智能体的行为来避免智能体之间冲突的方式关注潜在解决方案路径周

表4 代价增长树搜索算法

Table 4 Algorithms of increasing cost tree search

算法	主要机制	特点
ICTS	将高层搜索和低层搜索结合在一起。高层搜索目的是找出所有智能体的单个最优路大小,低层搜索目的是对高层搜索的状态节点 $(c_1,c_2,\cdots,c_k)$ 进行验证是否存在一组最优解	混合策略显著优于独立强化学习方法
E-ICTS	一种两级算法,在高层次搜索递增成本树,在低层搜索 k 个MDD的所有时间重叠的动作,以找到一组 k 个不冲突路径	在不同类型地图和不同规模障碍环境中保持着较好的性能,泛化性好
CBICS	结合了CBS和ICTS的优势	比CBS和ICTS更快地找到解决方案

表5 其他集中式规划算法

Table 5 Other centralized planning algorithms

算法	主要机制	特点
MAPF-LNS	首先快速找到初始解决方案,然后通过大领域搜索(LNS)反复重新规划智能体子集。它通过随机破坏启发式生成重新规划的智能体子集,但并非所有的智能体子集都能显著提高解决方案的质量	求解速度快,适用于200个智能体规模环境
MAPF-ML-LNS	MAPF-LNS算法基础加以改进,使用机器学习(ML)来学习如何从子集集合中选择智能体子集,以便找到更高质量的解决方案	在改进解决方案的速度和质量方面均优于MAPF-LNS
MAPF-LNS2	从一组包含冲突的路径开始,重复选择一个冲突智能体子集并重新规划他们的路径以减少冲突的数量,直到路径变得无冲突	MAPF-LNS2与最先进的随机重启的优先规划和EECBS相比,运行速度明显较快。MAPF-LNS2在大多数MAPF实例中,运行时间限制仅为5 min
CTRMs	使每个智能体能够以考虑其他智能体的行为来避免智能体之间冲突的方式关注潜在解决方案路径周围的重要位置	CTRMs开发了一种机器学习的方法,从相关问题实例和解决方案中学习生成模型
HMAPP	将环境划分为多个区域并将MAPF实例分解为每个区域的较小MAPF子实例来创建空间层次结构	能够在各种地图上获得更好的解决方案
Co-CBS	智能体不干扰彼此,而且还帮助彼此实现目标	自然地模拟了许多现实世界需要协作才能完成任务的应用程序
ASB-ECBS	一种自适应智能体特定的次优边界方法(ASB-ECBS),可以静态或者动态执行,可以根据单个智能体的要求分配次优界限	显著改善了运行时间,同时减少了搜索空间
PIBT	通过允许智能体具有临时优先级并限制智能体在树形路径的移动	使仓库分拣任务能够在具有一些树形路径的环境中执行而不会出现死锁
G-CBS	不同于经典MAPF假设,提出具有异构智能体MAPF概念	MAPF推广到异构智能体(G-MAPF)的情况,且不会导致显著的额外开销

围的重要位置。CTRMs开发了一种机器学习的方法,从相关问题实例和解决方案中学习生成模型,然后使用学习到的模型对CTRMs的顶点进行采样,以获得新的问题实例。文献[49]提出一种交通流预测算法来估计未来视野中的机器人密度分布,并在扇区级路径规划中考虑这些信息,通过平衡整个环境的交通流量来减少机器人拥堵并提高大型仓库工作效率。以最优方式求解MAPF是NP-Hard问题,因此现有最优和有界次优MAPF求解器通常无法扩展到大型MAPF实例。文献[50]提出一种新颖的MAPF求解器,分层多智能体路径规划器(HMAPP),它通过将环境划分为多个区域并将MAPF实例分解为每个区域的较小MAPF子实例来创建空间层次结构。对每个子实例,它使用有界次优MAPF求解器对其进行求解。HMAPP能够在各种地图上获得更好的解决方案。MAPF经典方法假设智能体在执行过程中盲目遵循无碰撞路径,然而在实际执行过程中可能会发生冲突情况,文献[51]提出一种鲁棒性概念,它使智能体在延迟的情况下转移到潜在路径。这种方法与之前的 K 步延迟计划相比,到达目标的概率要高很多。文献[52]提出基于协作冲突搜索的合作MAPF算法(Co-CBS),智能体不干扰彼此,而且还帮助彼此实现目标。这个扩展自然地模拟了许多现实世界需要协作才能完成任务的应用程序。文献[53]针对大型仓库分拣机器人控制问题,提出一种扩展的带有回溯优先级集成方法(PIBT),使其更适用于一般环境。它提出的方法使仓库分拣任务能够在具有一些树形路径的环境中执行而不会出现死锁,同时保留PIBT的特性,它通过允许智能体具有临时优先级并限制智能体在树形路径的移动来做到这一点。经典MAPF问题假设所有智能体都是同质的,具有固定的大小和行为。然而,实际应用中有些智能体是异构的。文献[54]将MAPF推广到异构智能体(G-MAPF)的情况,提出G-CBS算法,它不会导致显著的额外开销。

3 分布式执行算法

近年来,经典的MAPF算法已经能够解决大部分路径规划问题。然而这些问题的前提假设都是中央规划器掌握完整的地图信息和所有智能体的位置等信息,并且集中式方法需要收集地图信息和所有智能体的信息以规划最优路径,导致消耗大量的计算资源。随着技术的发展,去中心化的方法越来越流行,智能体在与环境交互过程中,通过和一定距离范围内的其他智能体通信来规划路径,泛化性较好,可以扩展到大规模智能体的环境。本章先对强化学习、深度学习和深度强化学习进行概述,然后介绍深度强化学习用于解决多智能体路径规划的算法,分析现有算法的优缺点,并提出了现有算法面临的挑战。

3.1 深度强化学习背景

近年来,深度强化学习(deep reinforcement learning,DRL)[55-56]取得显著成绩,这导致了DRL的应用场景和方法的数量激增。最近的研究也从单智能体到多智能体系统。尽管在复杂的多智能体领域面临许多挑战,但在一些复杂的游戏领域取得了许多成功,如围棋[57-58]、扑

克[59-60]、DOTA2[61]和星际争霸(StarCraft)[62]。这些领域的成功都依赖于两个技术的组合:强化学习(reinforcement learning,RL)和深度学习(deep learning,DL)。

3.1.1 强化学习

RL是一项通过不断试错来学习的技术。智能体通过一系列的步数与环境进行交互,在每一步,智能体基于当前的策略来获取环境状态,到达下一个状态并获得该动作奖励,智能体的目标是更新自己的策略以最大化累计奖励。如果环境满足马尔可夫性质,如公式(4)所示,RL可以建模为一个马尔可夫决策过程(Markov decision process,MDP)[63]。

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

其中 s_t 表示时间步 t 时的状态。

MDP可以用公式(5)来表示:

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

其中,S 表示状态空间 $\left(s_t\in S\right)$,A 表示动作空间,$a_t\in A$,R 表示奖励空间 $\left(r_t\in R\right)$,ρ 表示状态转移矩阵 $\left(\rho_{SS'}=P\left[s_{t+1}=s'|s_t=s\right]\right)$,$\gamma$ 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。

RL可分为以下基本方法:

(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。两种最主要的方法是值迭代和Q-learning[64],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。Q-learning通过贝尔曼方程[65]来更新动作值。

(2)基于策略方法(policy-based methods)[66]。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REINFORCE[67],该方法的模型是关于 $\theta\left(\pi_\theta\left(a|s\right)\right)$ 的函数,通过梯度上升来更新参数时收益最大化。

(3)演员评论家方法(actor-critic methods,AC)[68]。AC是一种混合方法,它学习策略和值函数。该算法由两个共享参数的模型组成:

Critic:更新值函数的参数(状态值函数或者动作值函数)。

Actor:更新策略参数。

该方法依赖于时序差分法(temporal difference,TD):计算动作的值的TD误差来更新动作值参数,图2表示了AC方法。

3.1.2 深度学习

深度学习是机器学习的一个领域,它使用含有几个隐藏层的神经网络(neural networks)从数据中提取有用的模型。它在人脸识别、图像识别、语言识别等领域都取得不错的效果。深度学习中最重要三种网络模型

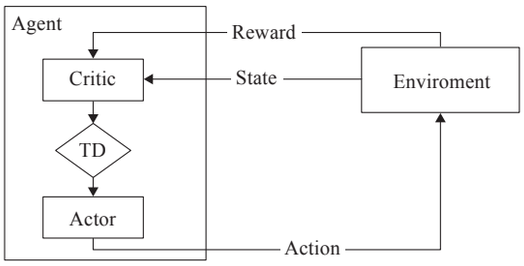

图2　AC强化学习智能体

Fig.2　Actor-Critic reinforcement larning agent

是卷积神经网络(convolutional neural network,CNN)、循环神经网络(recurrent neural network,RNN)和生成对抗网络(generative adversarial network,GAN)。

CNN在计算机视觉和图像处理等领域取得不错效果。它利用“卷积”的图像相关特征自动从数据中学习。通过增加CNN网络的宽带和深度可以使性能得到提高,但同时也会导致梯度消失等问题。

RNN在自然语言处理和时间序列方面表现较好的性能。它的主要优点是可以处理任意长度的输入,计算过程中考虑历史信息。虽然RNN是一个非常有用的工具但是也有缺点,计算速度慢并且难以从较早的过去获取信息。

GAN主要由两部分组成:一个生成模型G和一个判别模型D。这两个模型互相博弈学习产生较好的输出。判别模型估计来自输入数据样本的概率,生成模型生成一个接近真实样本的新样本。

3.1.3 深度强化学习

深度强化学习利用神经网络来估计值函数和策略。下面介绍几种主要单智能体强化学习算法。

基于值函数:deep Q-network(DQN)[69-70]使用神经网络来估计动作值函数,用神经网络代替Q-learning的Q表格。DQN采用经验回放(experience replay buffer)来降低数据之间的关联性,将与环境互动的数据 $D=e_1,e_2,\cdots,e_T$,其中 $e_t=\left(s_t,a_t,r_t,s_{t+1}\right)$,$t\in\left[1,T\right]$,存放在经验回放池中。在每次迭代中,使用基于TD的损失函数见公式(6)所示:

$$L_i\left(\theta_i\right)=E_{(s,a,r,s')\sim D}\left[\left(r+\gamma\max_{a'}Q\left(s',a',\theta_i^-\right)-Q\left(s,a,\theta_i\right)\right)^2\right] \tag{6}$$

其中 θ_i^-、θ_i 分别表示目标网络和Q网络的参数。DQN的主要缺点是只能处理低维离散的动作空间。此后,DQN算法演变出很多改进变体,如两个不同神经网络的Double DQN[71],动作值优势函数的Dueling DQN[72],优先经验重放算法[73],添加网络噪声以提升搜索的NoisyNet DQN[74],以及综合以上算法的Rainbow DQN[75]。

基于策略函数:深度确定策略梯度(deep deterministic policy gradient,DDPG)[76]结合了DQN和AC方法来学习策DDPG包含四个神经网络:当前Actor网络、目标Actor网络、当前Critic网络、目标Actor网络。

DDPG的目标是维护将状态映射到动作的Actor函数，并学习估计状态动作值的Critic函数。

基于策略的经典深度强化学习算法还有近端策略优化算法PPO[77]、置信域策略优化算法TRPO[78]、分布式策略梯度PPO算法[79]、异步优势算法(A3C)[80]、双延迟确定性策略梯度算法(TD3)[81]。

单智能体强化学习主要算法在表6总结。

3.1.4　多智能体深度强化学习

RL已经应用到许多具有挑战性的问题，如Alpha Go和Alpha Zero。然而实际应用场景中往往需要多个智能体之间的通信协调与合作。目前多智能体深度强化学习(multi-agent DRL，MADRL)正成为研究的热点。

分布式部分可观测马尔可夫决策过程：一个完全合作的多智能体强化学习任务可以用分布式部分可观测马尔可夫决策过程(Dec-POMDP)[82]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，$P(s'|s,u):S\times U\times S\to[0,1]$ 表示状态 s 下采取联合动作 u 转移到 s' 状态转移概率；$r(s,u):S\times U\to R$ 表示奖励函数；$z\in Z$ 表示每个智能体的观察值由 $O(s,a):S\times A\to Z$ 来描述；$\gamma\in(0,1)$ 表示折扣因子。

基于值函数分解的算法：将联合值函数分解为每个智能体的值函数。代表算法有VDN[83]、QMIX[84]、QTRAN[85]、NDQ[86]、CollaQ[87]、IQL[88]、QPLEX[89]、QPD[90]。

基于策略函数的算法：每个智能体通过自身观察的历史信息学习策略，大多采用集中式训练分布式执行方法，代表算法有MADDPG[91]、COMA[92]、IPPO[93]、MAPPO[94]、MAAC[95]。MADRL主要算法在表7总结。

3.2　强化学习在多智能体路径规划问题的适用性

探索RL在MAPF上的应用，需要对MAPF问题进行建模，并且该建模环境允许使用RL方法和传统集中式规划算法，以进行算法对比分析。

3.2.1　环境

MAPF所需环境需要满足适合训练RL智能体和集中式规划算法，以便对各种算法性能进行对比。2020年的NeurIPS Flatland挑战赛中使用的火车调度模拟环境[96]就是一种用于MAPF的简单环境，通常用于测试和比较各种MAPF算法。环境可视化如图3所示。

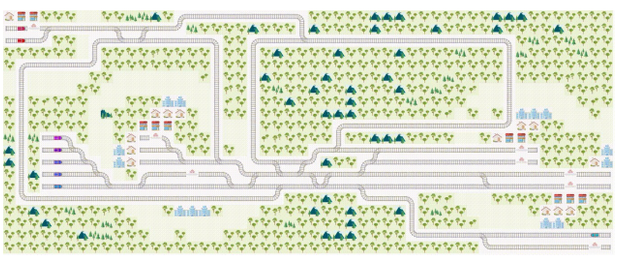

图3　7辆火车到达不同火车站的简单环境

Fig.3　Simple environment in which 7 trains arrive at different railway stations

在铁轨上的7辆不同火车从不同位置出发，去往不同的火车站。每个火车相当于一个智能体，火车站相当于目标位置。

状态空间：每个智能体有一个位置 (x,y)，其中 $x\in[0,w-1]$，$y\in[0,h-1]$，w 是网格环境的宽度，h 是网格环境的高度。每个智能体的位置和目的地以及路径组成状态空间。

动作空间：每个智能体的运动时间离散为每个时间步长，智能体在任一时间步长可采取行动 a，$a\in[0,3]$，分别表示智能体等待、向上、向下和向前运动。

奖励函数：每个智能体接受局部奖励和全局奖励。在每个时间步长内，如果智能体沿着铁轨移动或者停止，奖励为 r_l，如果智能体达到目的地，则奖励为 r_m，如果智能体之间发生碰撞，则奖励为 r_n，如果所有智能体

表6　单智能体强化学习算法对比

Table 6　Comparison of single agent reinforcement learning algorithms

算法	主要机制	特点
DQN	在Q-learning的基础上引入了神经网络Q-Network和经验回放池。Q-Network拟合Q值，有效解决高维状态动作下Q-table无法存储的问题，经验回放打破数据之间的关联性，提高样本利用率	解决了高维状态输入低维动作输出题。对Q值估计过高，训练不稳定
Double DQN	引入两个Q网络，一个网络估计最大值的动作，另一个网络估计这个动作的Q值	避免Q值估计过高的问题
Dueling DQN	对DQN神经网络输出进行了优化，将 $Q(s,a)$ 分解成状态价值函数 $V(s)$ 和优势函数 $Advantage(s,a)$ 之和	大大提高Q值的更新速度
NoisyNet DQN	将高斯噪声添加到网络的最后层，通过权重网络的训练来更新参数	能够使用较小的计算成本，实现较优的结果
Rainbow DQN	综合了DQN的六种改进技术	适用性更广
DDPG	基于AC框架，采用Actor网络、Critic网络、Actor目标网络、Critic目标网络四个不同的神经网络	解决连续动作问题，输出的直接是一个动作。在小型任务上收敛很快
TRPO	解决策略参数变化过大的问题，保证每一步新的策略单调递增	收敛速度较快
PPO	基于AC框架，采用重要性采样技术和 N 步更新	输出每个动作的概率。训练稳定，稳健性较好
A3C	借用经验回放池思想并加以改进，使用多个线程与环境进行交互，并异步更新	降低了数据之间的相关性，收敛更快

表7　多智能体深度强化学习算法对比

Table 7　Comparison ofmulti-agent deep reinforcement learningalgorithms

算法	主要机制	特点
IQL	每个智能体使用DQN算法暴力的求解多智能体问题	环境不稳定,难以收敛,无法体现智能体之间互动影响
VDN	采用集中式训练,分散式执行框架。将每个智能体的Q网络累加求和去近似全局Q网络,每个智能体根据分解的Q函数贪婪选择动作	较完全分散式的IQL算法,算法收敛速度快,在一些简单任务中快速又高效
QMIX	提出用神经网络将局部Q网络与整体Q网络之间关系拟合成一个单调函数,在训练过程中额外使用了全局状态信息。	相比于VDN算法,能够表征复杂智能体局部Q网络和全局Q网络之间关系
QPLEX	分别对联合Q值和各个智能体的Q值使用Q=V+A进行分解。将IGM一致性转换为易于实现的优势函数取值范围约束	算法计算量显著减少,并且具有较好的扩展性
QTRAN	将原始的整体值函数Q映射到一个新的值函数Q,使得到的这两个函数的最优联合动作是等价的	算法在非单调任务上性能显著,但计算成本高,结构复杂
NDQ	在现有值分解算法上引入通信,智能体偶尔向其他智能体发送信息以实现协调,同时引入正则化	较大减少通信开销
CollaQ	为每个智能体求解近似最优委派奖励,加强智能体之间的合作	在星际争霸的一些极度困难的任务上有较好的表现
QPD	一种Q值路径分解方法,将全局Q值分解为局部Q值。使用积分梯度的方法,沿轨迹路径直接分解全局Q值,为智能体分配信度	在同质和异质的多智能体场景中均达到先进的性能
MADDPG	采用集中式训练,分布式执行框架。改进了经验回放记录的数据。对每个智能体学习多个策略,利用所有策略的整体效果进行优化	既能用于合作环境,又能用于竞争环境。不需要环境的动力学模型及特殊的通信需求
COMA	采用集中式训练,分布式执行框架。使用一个集中式的Critic网络,在训练过程中获取所有智能体的信息。采用反事实基线来解决信用分配问题。Critic网络能够对反事实基线进行高效的计算	主要针对多智能体协作任务和离散动作
IPPO	每个智能体使用PPO算法独立学习在自己的策略,其他智能体视为环境的一部分	在一些简单环境中收到较好效果

都到达目标位置,则全局奖励为 r_o ,最后总的奖励 $r_i(t)$ 为四者之和: $R_i(t)=r_l(t)+r_m(t)+r_n(t)+r_o(t)$ 。

目标函数:强化学习的目标就是每个智能体通过与环境互动采取动作来达到未来奖励之和最大化。从时间步长 $t+1$ 到时间步 T 的回报定义为 G_t ,用公式(7)表示:

$$G_t=\sum_{k=0}^{T-t-1}\gamma^k R_{t+k+1}=R_{t+1}+\gamma\left(\sum_{k=0}^{T-t-2}\gamma^k R_{t+1+k+1}\right)=R_{t+1}+\gamma G_{t+1} \tag{7}$$

3.2.2　多智能体深度强化学习算法选择与实现

MADRL在MAPF上的算法实现主要有三种方式:(1)独立学习,每个智能体把其他智能体当做环境的一部分独立的使用单智能体RL算法来与环境互动执行动作,以使奖励最大化;(2)集中式训练分散式执行,这种方法有效解决信用分配问题,并且使智能体有效学会协作;(3)差异化通信,这种方法区别于完全分散式执行算法的隐式协调,可以直接通信,从而可以共享智能体之间的行动意图。

3.3　基于强化学习的多智能体路径规划方法研究进展

使用RL方法解决MAPF问题面临的许多挑战,例如环境奖励稀疏、环境动态复杂等。任何一种强化学习算法直接应用于MAPF问题都会出现学习速度慢和学习质量不高的问题。针对以上问题,研究人员采用了各种组合技术对基于强化学习的MAPF方法进行改进,使得强化学习的MAPF方法能够扩展到上千个智能体的环境,并且求解的质量和效率得到大大的提高。按照改进技术的特点,大致将基于RL的MAPF方法分为专家演示型、改进通信性和任务分解型三类,不同类型算法的优缺点总结于表8。

3.3.1　专家演示型

专家演示型方法主要采用强化学习和模仿学习(imitation learning,IL)和相结合的方法,这部分工作总结于表9。

PRIMAL(pathfinding via reinforcement and imitation multi-agent learning)[97],一种新的MAPF框架,它的算法原理如图4所示。在每一回合开始,随机选择是

表8　分布式执行算法对比

Table 8　Comparison of distributed execution algorithms

算法	优点	缺点
专家演示型	训练的策略可以扩展到较大规模智能体环境,学习速度快	使用集中式MAPF规划器生成专家演示,导致计算耗时
改进通信型	成功率较高,平均步长低,通信开销低	训练时间长,学习速度慢
任务分解型	规划效率、通用性和可扩展性好	结构复杂,实现较为困难

表9 专家演示型算法

Table 9 Expert demonstration algorithms

算法	主要机制	特点
PRIMAL	结合强化学习和模仿学习来训练完全分散的策略,其中智能体在部分可观察的环境中规划路径,同时智能体之间表现出隐式协调	学习到的策略可以扩展到1 000个智能体规模的环境
PRIMALc	通过奖励塑性和梯度裁剪,将通信引入PRIMAL,使模仿损失稳定地收敛到相对较低的值	将PRIMAL的工作由2D扩展到3D搜索空间
PRIMAL2	采用分布式学习框架,当智能体到达目标后立即被分配一个新的任务	在完成时间和接的质量与最先进的MAPF求解器比较都有了明显的提升,并且能扩展到2 000多个智能体
MAPPER	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	能够使用较小的计算成本,实现较优的结果
AB-MAPPER	在AC强化学习框架下引入注意力机制和BicNet相结合的AB-MAPPER算法。在这个框架中,一方面利用具有通信功能的BicNet来实现团队内部协调,另一方面,集中的评论家网络可以选择性地将注意力权重分配给周围的智能体	在动态拥挤的场景中,AB-MAPPER算法大大优于MAPPER算法
GLAS	结合了避免局部最小值的集中规划优势和可扩展的分布式执行的优势	在各种机器人和障碍物密度下都具有较高的成功率

基于RL或者基于IL进行学习,在基于RL学习中,在每个时间步长,每个智能体从环境中获取其观察值和奖励值并输入到神经网络,神经网络输出一个动作。不同智能体的动作按照随机顺序依次执行。基于IL的方法中,由经典的MAPF规划器作为专家经验,对所有智能体动作进行协调规划。它结合了RL和IL去学习完全去中心化的策略。在这个框架中,智能体处在一个部分可观察的环境中,智能体只能观察到有限的视野内的信息。智能体学习考虑其他智能体的位置对自己的影响,从而有利于整个集体而不仅仅是自己的最优路径。通过同时学习单个智能体路径(主要是采用RL的方法)和模仿集中式专家经验(IL),智能体最终学习到一个去中心化策略,同时包含了智能体之间的协调合作。最终学习到的策略可以扩展到更大规模智能体的环境中。文

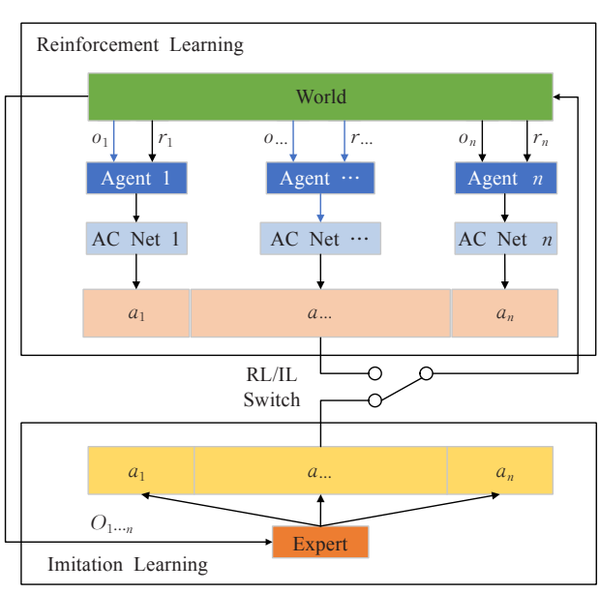

图4 RL/IL框架结构图

Fig.4 Structure of hybrid RL/IL

献[98]将PRIMAL的工作由2D扩展到3D搜索空间,提出PRIMALc算法,通过奖励塑性和梯度裁剪,将通信引入PRIMAL,使模仿损失稳定地收敛到相对较低的值。文献[99]提出PRIMAL2框架,该方法用于解决MAPF的一个变体:终身MAPF(lifelong MAPF,LMAPF),当智能体到达目标后立即被分配一个新的目标任务。PRIMAL2是一个分布式强化学习框架,智能体学习分散的策略,这样有利于智能体在一个部分可观察的环境中在线的规划路径。该方法通过构建新的局部观察和各种训练辅助工具将低密度的环境中学习到的经验扩展到高度结构化和高度约束的环境中。PRIMAL2在完成时间和接的质量与最先进的MAPF求解器比较都有了明显的提升,并且能扩展到2 000多个智能体。

文献[100]提出一个组合模型架构,该架构由一个局部观察值中提取特征的卷积神经网络(CNN)和在智能体之间交流这些特征信息的图神经网络(GNN)组成。该模型训练阶段模仿专家知识,然后用训练好的策略来在线规划路径。文献[101]提出一种基于进化强化学习方法MAPPER(multi-agent path planning with evolutionary reinforcement learning in mixed dynamic environments)在动态混合环境中来有效学习路径规划策略。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造成算法性能下降,MAPPER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。该文献的主要贡献是提出一种基于图像的表示方法来提高智能体处理不同障碍的鲁棒性和一种基于进化的MADRL方法来提高训练过程中的训练效率。文献[102]在MAPPER算法上进行了改进,在AC强化学习框架下引入注意力机制和BicNet[103]相结合的AB-MAPPER算法。在这个框架中,一方面利用具有通信功能的BicNet来实现团队内部协调,另一方面,集中

的评论家网络可以选择性地将注意力权重分配给周围的智能体。在动态拥挤的场景中,AB-MAPPER算法性能显著优于MAPPER算法。文献[104]提出GLAS(global-to-local autonomy synthesis)算法结合了避免局部最小值的集中规划优势和可扩展的分布式执行的优势。GLAS主要有三部分组成:(1)使用全局规划器生成演示轨迹并从中提取局部观察结果;(2)使用模仿学习来训练高效的分散策略;(3)引入可微分的安全模块,以确保智能体无碰撞操纵。

3.3.2　改进通信型

在高密度智能体的MAPF中,系统需要多个智能体在环境不完全可知的情况下相互关联,这就需要智能体之间通信,相关工作总结于表10。

在大规模智能体的MAPF中,为了加强协调与合作,智能体之间往往会进行通信,多智能体强化学习往往采用广播通信的方式,信息被传播到一定范围内的其他所有智能体。经验表明,这种广播通信的方法与非通信方法相比有较大的改进,但缺点是广播通信需要大量的带宽,并在实践中引起额外的系统开销和延迟。在广播通信中,并不是每个智能体的信息都对中央智能体有用。文献[105]提出一种消息感知图注意网络(message-aware graph attention networks,MAGAT)。近年来,图神经网络(GNN)在分布式多智能体系统中学习同行策略的能力越来越流行,然而一般的GNN依赖于简单的信息聚合机制,这种机制阻止了智能体对重要信息的优先排序。MAGAT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,并且能够很好解决之前未解决的MAPF实例,在大规模智能体的实例中,比基准成功率提高了47%。主要贡献是:(1)将图神经网络和一键查询机制相结合,提高智能体之间通信的有效性;(2)通过减少共享特征的大小来减少通信宽带以提高通信的有效性;(3)在小地图模型上训练策略,在大地图上进行测试,证明了此模型具有较好的泛化性和更高的效率。

文献[106]提出一种请求应答机制(DCC),其训练过程如图5所示。选择通信流程共分为四个模块:第一

表10　改进通信型算法

Table 10　Improved communication algorithms

算法	主要机制	特点
PICO	将隐式规划优先级结合到分散式MARL框架中,可以利用隐式优先级学习模块形成动态通信拓扑,从而建立有效的避碰机制	成功率更高,学习速度更快
DCC	智能体选择附近能改变自身策略的智能体进行通信	提高通信效率,减小通信带宽
DHC	将通信与深度Q学习相结合	成功率高,平均步长小
MAGAT	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	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

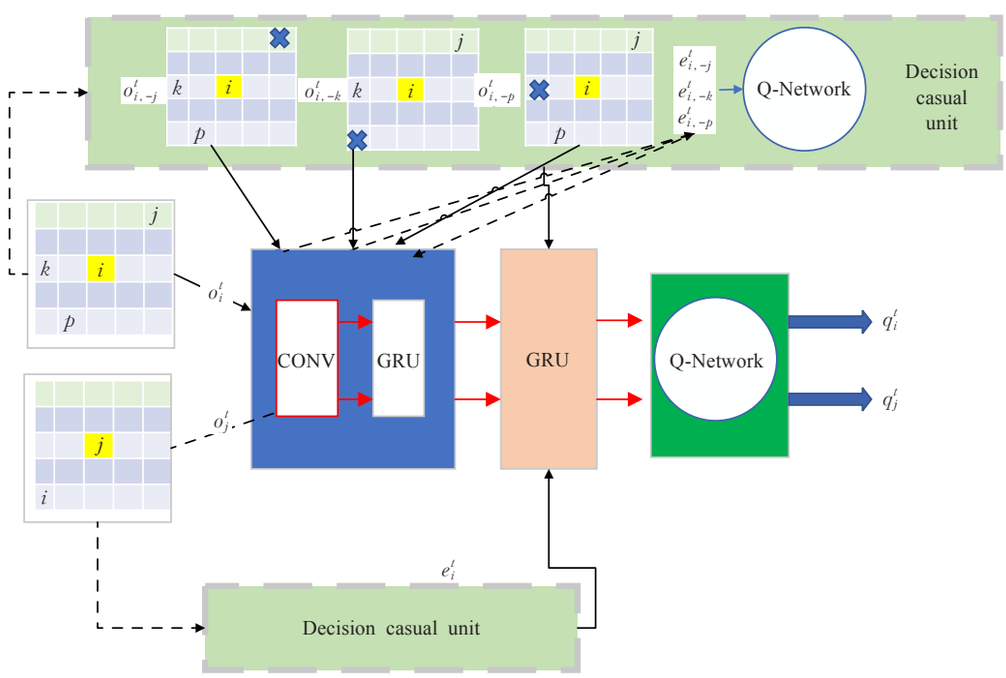

图5　选择通信流程图

Fig.5　System flow of decision communication

个模块是局部观察输入,智能体在每个时间步获得一个6×6的方格形状作为输入;第二个模块是观察编码,智能体的局部观察输入通过卷积网络GRU进行编码;第三个模块是选择判断单元,观察编码修改局部观察后输入到Q网络中,比较去掉该智能体和有该智能体的Q值变化来判断邻居智能体的信息是否对自己有用;第四个单元是通信模块,智能体选择通信后,将信息进行聚合,随后输入Q网络得到下一步的动作。文献[107]将通信与深度Q学习相结合,为MAPF提供了一种新的基于RL的方法(DHC),其中智能体通过图卷积网络实现协作。为了指导RL算法处理面向目标的长期任务,DHC嵌入了来自单一来源的最短路径的潜在选择作为启发式指导。在障碍物密度大的环境中的实验表明DHC方法的成功率高,平均步长小。基于RL的MAPF算法在高密度智能体环境中可能会产生更多的顶点冲突,从而导致成功率低或者学习时间更长,文献[108]提出一种优先通信学习方法(PICO),该方法将隐式规划优先级结合到分散式MARL框架中,可以利用隐式优先级学习模块形成动态通信拓扑,从而建立有效的避碰机制。PICO包括两个阶段,即隐式优先学习阶段和优先通信学习阶段。在隐式优先学习中,PICO构建一个辅助模仿学习任务,通过模仿经典来预测每个智能体的局部优先级。在优先通信中,PICO获得本地优先级用于生成有智能体集群组成的时变通信拓扑,每个智能体都可以通过接受到的消息来学习减少碰撞的策略。

3.3.3 任务分解型

大型动态环境的MAPF是一个非常具有挑战性的问题,因为智能体需要有效达到目标,同时避免与其他智能体或动态对象的冲突。解决这一挑战的重要方法是将任务进行分解,相关工作总结于表11。

文献[109]提出一种混合策略方法(HPL),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RL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,来设计智能体观察值到动作的映射,最后将学习到达目标策略和避免碰撞策略混合为一个策略。接下来,引入策略混个机制,最终得到一个单一的混合策略,该策略允许每个智能体表现出两类行为-个人行为(到达目标)和合作行为(避免与其他智能体发生冲突),其训练框架流程图如图6所示。

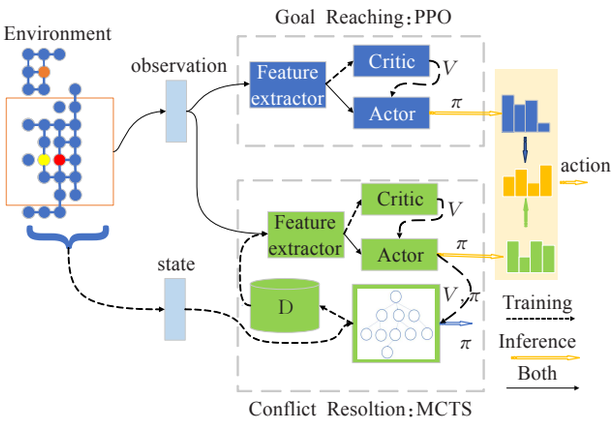

图6 混合策略方法

Fig.6 Hybrid policy approach

文献[110]提出G2RL算法(globally guided reinforcement learning approach)。该算法改进了经典算法中对动态障碍规划效率不高的缺陷,此外,该算法引入了一种新的奖励结构,具有良好的泛化能力,优于现有的分布式方法。该算法的主要贡献是:(1)提出一种分层框架,结合了全局规划和局部基于RL的规划以利于动态环境中学习到端到端策略。局部RL规划器利用局部观察信息来学习避免潜在的碰撞和不必要的绕行策略。(2)提出新的奖励结构,提供密集的奖励而不要求智能体在每一步严格遵守全局规划,以此激励智能体探索更多有潜力的路径。(3)该算法在不同类型地图和不同规模障碍环境中保持着较好的性能,具有较好的泛化性。文献[111]使用视觉数据的多机器人协作导航任务构建了RL框架(VRL),利用深度学习技术从原始视觉观察中提取高维特征,并将它们压缩成有效的表示。其次使用图神经网络来学习相邻机器人之间的信息共享和聚合,以实现有效的局部运动协调。该方法在大型机器人网络和大型环境中具有很好的可扩展性。

3.3.4 其他基于RL的MAPF算法

其他基于RL的MAPF算法主要有:(1)针对MAPF环境的特点,在MADRL基础上进行改进的MAPF算

表11 任务分解型算法

Table 11 Task decomposition algorithms

算法	主要机制	特点
HPL	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RL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,来设计智能体观察值到动作的映射,最后将学习到达目标策略和避免碰撞策略混合为一个策略	混合策略显著优于独立的强化学习方法。
G2RL	结合了全局规划和局部基于RL的规划以利于动态环境中学习到端到端策略。提出新的奖励结构,提供密集的奖励而不要求智能体在每一步严格遵守全局规划,以此激励智能体探索更多有潜力的路径	在不同类型地图和不同规模障碍环境中保持着较好的性能,具有较好的泛化性
VRL	利用深度学习技术从原始视觉观察中提取高维特征,并将他们压缩成有效的表示。其次使用图神经网络来学习相邻机器人之间的信息共享和聚合,以实现有效的局部运动协调	在大型机器人网络和大型环境中具有很好的可扩展性

法;(2)将基于RL的MAPF算法和传统算法或者其他领域先进技术相结合的算法;(3)针对MAPF应用在机场调度、自动仓库和无人机等实际应用场景,如何处理实际约束而提出的解决方案。

尽管基于RL的MAPF算法提高了算法的扩展性,但解决组合域仍然具有挑战性,因为智能体的随机探索不太可能产生有用的奖励信号,文献[112]将知识编译与RL相结合,将知识编译集成到基于策略梯度和Q值的各种MARL算法中,所得到的算法在样本复杂性和解决方案质量方面都显著优于原始算法。文献[113]从全局静态规划和局部动态规划两个方面分析无人机路径规划,在A^*算法基础上,改进搜索策略、步长和代价函数,从而缩短规划时间,大大提高算法的执行效率。此外,在Q-learning的探索机制中加入动态探索因子,解决了Q-learning的探索困境,以适应无人机的局部动态路径调整。将两者结合形成全局和局部相混合的无人机路径规划算法。文献[114]采用多步前进树搜索方法来做出有效的决策,随着智能体数量的增加,以更短路径长度和求解时间优于原有算法。文献[115]提出了一种改进自动化仓库中多个移动机器人学习路径的稀疏奖励问题的双分段奖励模型,使学习高效稳定的进行,该文献作为一个实例案例研究,具有较大意义。文献[116]采用无模型的在线Q学习算法,多个智能体重复"探索-学习-利用"过程,积累历史经验评估动作策略并优化决策,完成未知环境下的多智能体路径规划任务。文献[117]提出将随机探索与Q表探索相结合,在探索未知路径和利用已有路径中进行协调,提高探索效率。文献[118]提出一种分层强化学习及人工势场的多智能体路径规划算法,利用分层强化学习方法的无环境模型学习以及局部更新能力将策略更新过程限制在规模较小的局部空间或维度较低的高层空间上,提高算法的性能。文献[119]提出基于惯性权重的鸡群优化算法,有效解决传统路径规划算法收敛精度不高、容易陷入局部最优的问题。文献[120]针对智能体添加通信通道时智能体体输出大小随消息位数呈指数增长的限制,提出一个独立的按位消息策略参数化,允许智能体输出大小随信息内容线性缩放。这个改进显著提高了样本效率并导致改进最终策略。文献[121]提出一种在MARL背景下通过反向传播学习稀疏离散通信的方法,激励智能体尽可能少地进行通信,同时仍然获得高回报。文献[122]假设智能体在任意有向图上运行,而不是通常假设的网格世界,这扩展可对环境无法以网格形状建模的用例的支持。

3.4 基于RL的MAPF算法的挑战

目前,基于RL的MAPF面临大量挑战[123]。这些挑战来源于RL本身的特点以及外部环境。在实现基于RL的MAPF算法应用到实际场景中还要考虑许多因素。本节主要分析和总结这些挑战,并对未来的研究方向给出建议。

(1)奖励稀疏问题:多智能体路径规划过程通常是目标驱动的,环境给予的奖励通常在智能体到达目标终点。在环境尺寸较大的环境中,智能体很难获得最终的奖励信号。此外,为了促进智能体快速到达目标,给予每个时间步负奖励,但这也导致智能体在长期负奖励信号下产生异常的动作,例如智能体在原地保持不动。稀疏的奖励问题还会导致学习效率低下和收敛速度慢。解决奖励稀疏问题的主要方法有好奇心驱动、奖励塑形和分层强化学习等。好奇心驱动[124-126]主要思想是构建内在好奇心模块,以从环境中提取额外的内在奖励信号,鼓励智能体进行更有效的探索。奖励塑造[127]的主要思想是手动调整和修改智能体在不同状态下的奖励信号。奖励塑造更为直观,但是高度依赖于专家经验。分层强化学习[128]的主要思想是将任务分解为多个离散或连续且易于解决的子任务,然后分而治之。

(2)样本效率低:RL智能体每次面对新的任务,都要从头开始学习。在训练初期,过多的无用数据导致智能体低效的学习和更新策略。因此,如何更好地探索有价值的策略和提高有价值的经验采样效率是研究的热点方向,也是提高基于RL的MAPF算法性能的关键。文献[129-130]将优先经验重放技术应用与路径规划。文献[131]构造无监督表示作为辅助任务来提高样本效率。文献[132]提出自我预测表示,训练智能体来预测自己的潜在状态表示到未来的多个步骤,提高了在有限的交互中来收集数据的能力。

(3)环境动态复杂:多智能体路径规划任务需要多个智能体同时参与,且智能体之间需要相互配合并且决策的结果会相互影响。在多智能体系统中,各个智能体需要在环境不完全可知的情况下互相关联完成任务。在MAPF场景中的环境是复杂的、动态的。这些特性给学习过程带来很大的困难,例如,随着智能体数量的增长,联合状态及动作空间的规模呈现出指数增长;多个智能体同时学习,每个智能体策略改变都会影响其他智能体策略改变。针对上述困难,可以在动态环境中借助一些辅助信息弥补其不可见信息,从而提高学习的效率。为了达到这个目的,研究者们提出一些方法,例如,智能体之间的通信,即智能体通过发送和接受抽象的通信信息来分析环境中其他智能体的情况从而协调各自的策略。文献[133]提出使用注意力通信模型学习何时需要通信,以及如何整合共享信息以进行合作决策。文献[134]提出独立推断通信,使智能体能够学习智能体与智能体通信的先验知识。

4 算法小结

经典的集中式规划算法是目前最常用的也是效率最高的算法,如2020年的NeurIPS Flatland挑战赛(一种火车调度大赛),获得比赛第一名的是南加州大学的

表12 MAPF算法对比

Table 12 Comparison of MAPF algorithms

算法	优点	缺点
集中式规划算法	中央规划器对所有智能体进行规划,对于固定环境和小规模智能体环境,规划速度快,规划质量高	当智能体数量增大和环境更加动态复杂时,受到搜索空间大小限制,重新规划比较耗时,可扩展性差
分布式执行算法	每个智能体根据当前观察来独立执行动作,能够较好地扩展到大规模环境中,能够实时地处理路径重新规划问题	在静态小规模环境中的规划速度和效率比集中式规划算法低,学习时间长

李娇阳团队,她们使用的是经典的基于搜索的MAPF算法,能同时对上千辆火车进行高效调度。这种算法特点是在解决固定环境和智能体密度小障碍物密度低的MAPF问题速度快。基于多智能体深度强化学习的分布式执行算法在实时解决路径重新规划问题上展示了较大的潜力,缺点是训练时间较长。用集中式规划算法作为专家知识来加速强化学习的训练的方法在多智能体路径规划问题上收到较好效果。MAPF算法对比分析见表12所列。

5 研究展望

近年来,解决MAPF问题最常见的方法是集中式规划算法,但随着AlphaGo和AlphaZero的横空出世,采用强化学习方法来解决MAPF中的路径冲突问题成为研究热点。2020年NeurIPS Flatland挑战赛[96],虽然提交最好算法是经典集中式规划算法,但发现了许多有前途的基于MADRL的MAPF算法,使用RL方法实现了多样的协调机制。未来的研究方向归纳为以下3个方面:

(1)为所有智能体在任意环境中规划最优路径是NP-hard问题,因此使用实时的MAPF算法并不保证所有的智能体都成功规划路径。可以通过实验来确定在一组具有代表性的MAPF实例中最好的MAPF算法,但在什么时候使用哪个MAPF算法可以做得更好,在解决一些实际环境中必须快速做出决定,可以使用机器学习中的分类算法来定义一个快速计算映射,从MAPF实例的特征到性能最好的MAPF算法,以提高完成率,而不是使用单一算法解决每个问题。

(2)对于一个实际的MAPF问题,没有那种算法最有效。基于A^*和规约算法对于智能体密度小的环境最有效,CBS和ICTS算法对大型地图环境最有效,基于强化学习的分布式执行算法对部分可观察的环境最有效。未来工作的一个具有吸引力的方向是创建混合算法,以促进不同MAPF求解器的优势互补。

(3)现在MAPF问题假设智能体的动作是离散的,只有前后左右和等待五个动作,且没有考虑智能体运动的速度以及智能体体力消耗的因素,而真实世界的MAPF问题的智能体大多都是连续动作的,且智能体运动速度是连续变化,如何将离散动作升级到连续动作和节省智能体运动体力是未来工作的一个研究方向。

6 结束语

本文介绍了经典的MAPF集中式规划算法和人工智能领域兴起的基于强化学习的MAPF分布式执行算法。经典算法在求解静态环境的MAPF问题达到较高的求解速度和质量,基于RL的MAPF算法在实时重新再规划的场景中具有较好的扩展性。下一步研究工作的重点是运用模仿学习和强化学习的方法将两种类型算法相结合,以提高MAPF算法的效率和适用性。

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]//Proceedings of 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,2017:270-272.

[4] MOHANTY S,NYGREN E,LAURENT F,et al.Flatland-RL:multi-agent reinforcement learning on trains[J].arXiv:2012.05893,2020.

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

[6] CARTUCHO J,VENTURA R,VELOSO M.Robust object recognition through symbiotic deep learning in mobile robots[C]//2018 IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS),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] SURYNEK P,FELNER A,STERN R,et al.An empirical 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,ŠVANCARA J,ŠKOPKOVÁ V,et al.Multi-agent path finding on real robots:first experience with

ozobots[C]//Ibero-American Conference on Artificial Intelligence.Cham:Springer,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] STERN R,STURTEVANT N R,FELNER A,et al.Multi-agent pathfinding:definitions,variants,and benchmarks[C]//Twelfth Annual Symposium on Combinatorial Search,2019.

[13] STERN R.Multi-agent path finding-an overview[M]//Artificial intelligence.Berlin:Springer-Verlag,2019:96-115.

[14] 刘庆周,吴锋.多智能体路径规划研究进展[J].计算机工程,2020,46(4):1-10.
LIU Q Z,WU F.Research progress of multi-agent path planning[J].Computer Engineering,2020,46(4):1-10.

[15] FERGUSON D,LIKHACHEV M,STENTZ A.A guide to heuristic-based path planning[C]//Proceedings of the International Workshop on Planning Under Uncertainty for Autonomous Systems,International Conference on Automated Planning and Scheduling(ICAPS),2005:9-18.

[16] STANDLEY T.Finding optimal solutions to cooperative pathfinding problems[C]//Proceedings of the AAAI Conference on Artificial Intelligence,2010:173-178.

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

[18] VIRMANI L,REN Z,RATHINAM S,et al.Subdimensional expansion using attention-based learning for multi-agent path finding[J].arXiv:2109.14695,2021.

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

[20] REN Z,RATHINAM S,CHOSET H.Loosely synchronized search for multi-agent path finding with asynchronous actions[C]//2021 IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS),2021:9714-9719.

[21] GOLDENBERG M,FELNER A,STURTEVANT N,et al.Optimal-generation variants of EPEA[C]//International Symposium on Combinatorial Search,2013.

[22] GOLDENBERG M,FELNER A,STERN R,et al.Enhanced partial expansion A[J].Journal of Artificial Intelligence Research,2014,50(2):141-187.

[23] SHARON G,STERN R,FELNER A,et al.Conflict-based search for optimal multi-agent pathfinding[J].Artificial Intelligence,2015,219(2):40-66.

[24] BOYARSKI E,FELNER A,STERN R,et al.ICBS:improved conflict-based search algorithm for multi-agent pathfinding[C]//Twenty-Fourth International Joint Conference on Artificial Intelligence,2015.

[25] GANGE G,HARABOR D,STUCKEY P J.Lazy CBS:implicit conflict-based search using lazy clause generation[C]//Proceedings of the International Conference on Automated Planning and Scheduling,2019:155-162.

[26] BOYARSKI E,FELNER A,LE BODIC P,et al.f-Aware conflict prioritization & improved heuristics for conflict-based search[C]//Proceedings of the AAAI Conference on Artificial Intelligence,2021:12241-12248.

[27] LI J,FELNER A,BOYARSKI E,et al.Improved heuristics for multi-agent path finding with conflict-based search[C]//International Joint Conference on Artificial Intelligence(IJCAI),2019:442-449.

[28] BARER M,SHARON G,STERN R,et al.Suboptimal variants of the conflict-based search algorithm for the multi-agent pathfinding problem[C]//Seventh Annual Symposium on Combinatorial Search,2014.

[29] CHAN S H,LI J,GANGE G,et al.ECBS with flex distribution for bounded-suboptimal multi-agent path finding[C]//Proceedings of the International Symposium on Combinatorial Search,2021:159-161.

[30] RAHMAN M,ALAM M A,ISLAM M M,et al.An adaptive agent-specific sub-optimal bounding approach for multi-agent path finding[J].IEEE Access,2022,10:22226-22237.

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

[32] HUANG T,DILKINA B,KOENING S.Learning to resolve conflicts for multi-agent path finding with conflict-based search[C]//AAAI Conference on Artificial Intelligence,2020.

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

[34] BOYARSKI E,FELNER A,HARABOR D,et al.Iterative-deepening conflict-based search[C]//Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence,2021:4084-4090.

[35] SHAARON G,STERN R,GOLDENBERG M,et al.The increasing cost tree search for optimal multi-agent pathfinding[J].Artificial Intelligence,2013,195:470-495.

[36] WALKER T T,STURTEVANT N R,FELNER A.Extended increasing cost tree search for non-unit cost domains[C]//

International Joint Conference on Artificial Intelligence (IJCAI),2018:534-540.

[37] WALKER T T,STURTEVANT N R,FELNER A,et al. Conflict-based increasing cost search[C]//Proceedings of the International Conference on Automated Planning and Scheduling,2021:385-395.

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

[39] SURYNEK P.Makespan optimal solving of cooperative path-finding via reductions to propositional satisfiability[J]. arXiv:1610.05452,2016.

[40] BARTAK R,ZHOU N F,STERN R,et al.Modeling and solving the multi-agent pathfinding problem in picat[C]// 2017 IEEE 29th International Conference on Tools with Artificial Intelligence(ICTAI),2017:959-966.

[41] ZHOU N F,KJELLERSTRAND H,FRUHMAN J.Constraint solving and planning with picat[M].[S.l.]:Springer International Publishing,2015.

[42] ERDEM E,KISA D G,OZTOK U,et al.A general formal framework for pathfinding problems with multiple agents[C]//Twenty-Seventh AAAI Conference on Artificial Intelligence,2013.

[43] SURYNEK P.Multi-agent path finding with continuous time viewed through satisfiability modulo theories(SMT)[J]. arXiv:1903.09820,2019.

[44] YU J,LAVALLE S M.Multi-agent path planning and network flow[M]//Algorithmic foundations of robotics X.Berlin,Heidelberg:Springer,2013:157-173.

[45] LI J,CHEN Z,HARABOR D,et al.Anytime multi-agent path finding via large neighborhood search[C]//International Joint Conference on Artificial Intelligence(IJCAI), 2021.

[46] HUANG T,LI J,KOENING S,et al.Anytime multi-agent path finding via machine learning-guided large neighborhood search[C]//Proceedings of the AAAI Conference on Artificial Intelligence,2022.

[47] LI J,CHEN Z,HARABOR D,et al.MAPF-LNS2:fast repairing for multi-agent path finding via large neighborhood search[C]//Proceedings of the AAAI Conference on Artificial Intelligence,2022.

[48] OKUMURA K,YONETANI R,NISHIMURA M,et al. CTRMs:learning to construct cooperative timed roadmaps for multi-agent path planning in continuous spaces[J]. arXiv:2201.09467,2022.

[49] LIU Z,WANG H,WEI H,et al.Prediction,planning,and coordination of thousand-warehousing-robot networks with motion and communication uncertainties[J].IEEE Transactions on Automation Science and Engineering,2020, 18(4):1705-1717.

[50] ZHANG H,YAO M,LIU Z,et al.A hierarchical approach to multi-agent path finding[C]//Proceedings of the International Symposium on Combinatorial Search,2021:209-211.

[51] NEKVINDA M,BARTAK R.Contingent planning for robust multi-agent path finding[C]//2021 IEEE 33rd International Conference on Tools with Artificial Intelligence(ICTAI), 2021:487-492.

[52] GRESHLER N,GORDON O,SALZMAN O,et al.Cooperative multi-agent path finding:beyond path planning and collision avoidance[C]//2021 International Symposium on Multi-Robot and Multi-Agent Systems(MRS),2021:20-28.

[53] FUJITANI Y,YAMAUCHI T,MIYASHITA Y,et al.Deadlock-free method for multi-agent pickup and delivery problem using priority inheritance with temporary priority[J].arXiv: 2205.12504,2022.

[54] ATZON D,ZAX Y,KIVITY E,et al.Generalizing multi-agent path finding for heterogeneous agents[C]//Thirteenth Annual Symposium on Combinatorial Search,2020.

[55] HERNANDEZ-LEAL P,KARTAL B,TAYLOR M E.A survey and critique of multiagent deep reinforcement learning[J].Autonomous Agents and Multi-Agent Systems,2019, 33(6):750-797.

[56] OTHMAN W,SHILOV N.Deep reinforcement learning for path planning by cooperative robots:existing approaches and challenges[C]//2021 28th Conference of Open Innovations Association(FRUCT),2021:349-357.

[57] SILVER D,HHUANG A,MADDISON C J,et al.Mastering the game of Go with deep neural networks and tree search[J].Nature,2016,529(7587):484-489.

[58] SILVER D,SCHIRITTWIESER J,SIMONYAN K,et al. Mastering the game of go without human knowledge[J]. Nature,2017,550(7676):354-359.

[59] MORAVCIK M,SCHMID M,BURCH N,et al.Deepstack: expert-level artificial intelligence in heads-up no-limit poker[J].Science,2017,356(6337):508-513.

[60] BROWN N,SANDHOLM T.Superhuman AI for heads-up no-limit poker:libratus beats top professionals[J].Science, 2018,359(6374):418-424.

[61] BERNER C,BROCKMAN G,CHAN B,et al.Dota 2 with large scale deep reinforcement learning[J].arXiv:1912. 06680,2019.

[62] VINYALS O,EWALDS T,BARTUNOV S,et al.Starcraft ii: a new challenge for reinforcement learning[J].arXiv:1708. 04782,2017.

[63] FILAR J,VRIEZE K.Competitive Markov decision processes[M].[S.l.]:Springer Science & Business Media,2012.

[64] JANG B,KIM M,HARERIMANA G,et al.Q-learning algorithms:a comprehensive classification and applications[J].IEEE Access,2019,7:133653-133667.

[65] O'DONOGHUE B,OSBAND I,MUNOS R,et al.The uncertainty bellman equation and exploration[C]//International Conference on Machine Learning,2018:3836-3845.

[66] THOMAS P S,BRUNSKILL E.Policy gradient methods for reinforcement learning with function approximation and action-dependent baselines[J].arXiv:1706.06643,2017.

[67] WIERING M A,VAN OTTERLO M.Reinforcement learning[J].Adaptation,Learning,and Optimization,2012,12(3):729-734.

[68] BHATNAGAR S,SUTTON R S,GHAVAMZADEH M,et al.Natural actor-critic algorithms[J].Automatica,2009,45(11):2471-2482.

[69] MNIH V,KAVUKCUOGLU K,SILVER D,et al.Playing atari with deep reinforcement learning[J].arXiv:1312.5602,2013.

[70] MNIH V,KAVUKCUOGLU K,SILVER D,et al.Human-level control through deep reinforcement learning[J].Nature,2015,518(7):529-533.

[71] VAN HASSELT H,GUEZ A,SILVER D.Deep reinforcement learning with double q-learning[C]//Proceedings of the AAAI Conference on Artificial Intelligence,2016.

[72] WANG Z,SCHAUL T,HESSEL M,et al.Dueling network architectures for deep reinforcement learning[C]//International Conference on Machine Learning,2016:1995-2003.

[73] SCHAUL T,QUAN J,ANTONOGLOU I,et al.Prioritized experience replay[J].arXiv:1511.05952,2015.

[74] FORTUNATO M,AZAR M G,PIOT B,et al.Noisy networks for exploration[J].arXiv:1706.10295,2017.

[75] HESSEL M,MODAYIL J,VAN-HASSELT H,et al.Rainbow:combining improvements in deep reinforcement learning[C]//Thirty-Second AAAI Conference on Artificial Intelligence,2018.

[76] LILLICRAP T P,HUNT J J,PRITZEL A,et al.Continuous control with deep reinforcement learning[J].arXiv:1509.02971,2015.

[77] SCHULMAN J,WOLSKI F,DHARIWAL P,et al.Proximal policy optimization algorithms[J].arXiv:1707.06347,2017.

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

[79] HEESS N,TB D,SRIRAM S,et al.Emergence of locomotion behaviours in rich environments[J].arXiv:1707.02286,2017.

[80] MNIH V,BADIA A P,MIRZA M,et al.Asynchronous methods for deep reinforcement learning[C]//International Conference on Machine Learning,2016:1928-1937.

[81] FUJIMOTO S,HOOF H,MEGER D.Addressing function approximation error in actor-critic methods[C]//International Conference on Machine Learning,2018:1587-1596.

[82] OLIEHOEK F A,SPAAN M T J,VLASSIS N.Optimal and approximate Q-value functions for decentralized POMDPs[J].Journal of Artificial Intelligence Research,2008,32:289-353.

[83] SUNEHAG P,LEVER G,GRUSLYS A,et al.Value-decomposition networks for cooperative multi-agent learning[J].arXiv:1706.05296,2017.

[84] RASHID T,SAMVELYAN M,SCHROEDER C,et al.Qmix:monotonic value function factorisation for deep multi-agent reinforcement learning[C]//International Conference on Machine Learning,2018:4295-4304.

[85] SON K,KIM D,KANG W J,et al.Qtran:learning to factorize with transformation for cooperative multi-agent reinforcement learning[C]//International Conference on Machine Learning,2019:5887-5896.

[86] WANG T,WANG J,ZHENG C,et al.Learning nearly decomposable value functions via communication minimization[J].arXiv:1910.05366,2019.

[87] ZHANG T,XU H,WANG X,et al.Multi-agent collaboration via reward attribution decomposition[J].arXiv:2010.08531,2020.

[88] 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.

[89] WANG J,REN Z,LIIU T,et al.Qplex:duplex dueling multi-agent q-learning[J].arXiv:2008.01062,2020.

[90] YANG Y,HAO J,CHEN G,et al.Q-value path decomposition for deep multiagent reinforcement learning[C]//International Conference on Machine Learning,2020:10706-10715.

[91] LOWE R,WU Y,TAMAR A,et al.Multi-agent actor-critic for mixed cooperative-competitive environments[J].arXiv:1706.02275,2017.

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

[93] DE WITT C S,GUPTA T,MAKOVIICHUK D,et al.Is independent learning all you need in the StarCraft multi-agent challenge[J].arXiv:2011.09533,2020.

[94] YU C,VELU A,VINITSKY E,et al.The surprising effectiveness of mappo in cooperative,multi-agent games[J].arXiv:2103.01955,2021.

[95] IQBAL S,SHA F.Actor-attention-critic for multi-agent

reinforcement learning[C]//International Conference on Machine Learning,2019:2961-2970.

[96] LAURENT F,SCHNEIDER M,SCHELLER C,et al.Flatland competition 2020:MAPF and MARL for efficient train coordination on a grid world[J].arXiv:2103.16511,2021.

[97] 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.

[98] ZHIYAO L,SARTORETTI G.Deep reinforcement learning based multiagent pathfinding[R].2020.

[99] DAMANI M,LUO Z,WENZEL E,et al.PRIMAL2:pathfinding via reinforcement and imitation multi-agent learning-lifelong[J].IEEE Robotics and Automation Letters,2021,6(2):2666-2673.

[100] LI Q,GAMA F,RIBEIRO A ,et al.Graph neural networks for decentralized multi-robot path planning[C]//2020 IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS),2020:11785-11792.

[101] 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),2020:11748-11754.

[102] GUAN H,GAO Y,ZHAO M,et al.AB-Mapper:attention and BicNet based multi-agent path finding for dynamic crowded environment[J].arXiv:2110.00760,2021.

[103] PENG P,WEN Y,YANG Y,et al.Multiagent bidirectionally-coordinated nets:emergence of human-level coordination in learning to play starcraft combat games[J].arXiv:1703.10069,2017.

[104] RIVIERE B,HONIG W,YUE Y,et al.Glas:global-to-local safe autonomy synthesis for multi-robot motion planning with end-to-end learning[J].IEEE Robotics and Automation Letters,2020,5(3):4249-4256.

[105] 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.

[106] MA Z,LUO Y,PAN J.Learning selective communication for multi-agent path finding[J].arXiv:2109.05413,2021.

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

[108] LI W,CHEN H,JIN B,et al.Multi-agent path finding with prioritized communication learning[J].arXiv:2202.03634,2022.

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

[110] 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.

[111] LIU Z,LIU Q,TANG L,et al.Visuomotor reinforcement learning for multirobot cooperative navigation[J].IEEE Transactions on Automation Science and Engineering,2021:1-12.

[112] LING J,CHANDAK K,KUMAR A.Integrating knowledge compilation with reinforcement learning for routes[C]//Proceedings of the International Conference on Automated Planning and Scheduling,2021:542-550.

[113] LI D,YIN W,WONG W E,et al.Quality-oriented hybrid path planning based on A* and Q-Learning for unmanned aerial vehicle[J].IEEE Access,2021,10:7664-7674.

[114] ZHANG Y,QIAN Y,YAO Y,et al.Learning to cooperate:application of deep reinforcement learning for online AGV path finding[C]//Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems,2020:2077-2079.

[115] LEE H,HONG J,JEONG J.MARL-based dual reward model on segmented actions for multiple mobile robots in automated warehouse environment[J].Applied Sciences,2022,12(9):4703.

[116] 王毅然,经小川,田涛,等.基于强化学习的多Agent路径规划方法研究[J].计算机应用与软件,2019,36(8):165-171.
WANG Y R,JING X C,TIAN T,et al.Multi-agent path planning based on reinforcement learning[J].Computer Applications and Software,2019,36(8):165-171.

[117] 陈思豪,赵成业,王超,等.基于强化学习的多智能体路径规划算法[C]//第32届中国过程控制会议(CPCC2021)论文集,2021:1619.
CHEN S H,ZHAO C Y,WANG C,et al.Multi agent path planning algorithm based on reinforcement learning[C]//32nd Chinese Process Control Conference(CPCC2021),2021:1619.

[118] 郑延斌,李波,安德宇,等.基于分层强化学习及人工势场的多Agent路径规划方法[J].计算机应用,2015,35(12):3491-3496.
ZHENG Y B,LI B,AN D Y,et al.Multi-agent path planning algorithm based on hierarchical reinforcement learning and artificial potential field[J].Journal of Computer Applications,2015,35(12):3491-3496.

[119] 张靖南.基于多智能体的群体路径规划研究[D].哈尔

滨:哈尔滨工程大学,2019.
ZHANG J N.Research on group path planning based on multi-agent[D].Harbin:Harbin Engineering University,2019.

[120] FREED B,SARTORETTI G,CHOSET H.Simultaneous policy and discrete communication learning for multi-agent cooperation[J].IEEE Robotics and Automation Letters,2020,5(2):2498-2505.

[121] FREED B,JAMES R,SARTORETTI G,et al.Sparse discrete communication learning for multi-agent cooperation through backpropagation[C]//2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),2020:7993-7998.

[122] VAN KNIPPENBERG M,HOLENDERSKI M,MENKOVSKI V.Time-constrained multi-agent path finding in non-lattice graphs with deep reinforcement learning[C]//Asian Conference on Machine Learning,2021:1317-1332.

[123] HE Z,WANG J,SONG C.A review of mobile robot motion planning methods:from classical motion planning workflows to reinforcement learning-based architectures[J].arXiv:2108.13619,2021.

[124] PATHAK D,AGRAWAL P,EFROS A A,et al.Curiosity-driven exploration by self-supervised prediction[C]//International Conference on Machine Learning,2017:2778-2787.

[125] ZHELO O,ZHANG J,TAI L,et al.Curiosity-driven exploration for mapless navigation with deep reinforcement learning[J].arXiv:1804.00456,2018.

[126] SHI H,SHI L,XU M,et al.End-to-end navigation strategy with deep reinforcement learning for mobile robots[J].IEEE Transactions on Industrial Informatics,2019,16(4):2393-2402.

[127] NG A Y,HARADA D,RUSSELL S.Policy invariance under reward transformations:theory and application to reward shaping[C]//Proceedings of ICML,1999,99:278-287.

[128] 赖俊,魏竞毅,陈希亮.分层强化学习综述[J].计算机工程与应用,2021,57(3):72-79.
LAI J,WEI J Y,CHEN X L.Overview of hierarchical reinforcement learning[J].Computer Engineering and Applications,2021,57(3):72-79.

[129] HU Z J,GAO X J,WAN K F,et al.Relevant experience learning:a deep reinforcement learning method for UAV autonomous motion planning in complex unknown environments[J].Chinese Journal of Aeronautics,2021,34(12):187-204.

[130] HE Z,DONG L,SUN C,et al.Reinforcement learning based multi-robot formation control under separation bearing orientation scheme[C]//2020 Chinese Automation Congress(CAC),2020:3792-3797.

[131] LADKIN M,SRINVAS A,ABBEEL P.CURL:contrastive unsupervised representations for reinforcement learning[C]//International Conference on Machine Learning,2020:5639-5650.

[132] SCHWARZER M,ANAND A,GOEL R,et al.Data-efficient reinforcement learning with self-predictive representations[J].arXiv:2007.05929,2020.

[133] JIANG J,LU Z.Learning attentional communication for multi-agent cooperation[C]//Advances in Neural Information Processing Systems,2018.

[134] DING Z,HUANG T,LU Z.Learning individually inferred communication for multi-agent cooperation[C]//Advances in Neural Information Processing Systems,2020:22069-22079.