doi:10.3772/j.issn.1002-0470.2024.10.007

基于图神经网络的多智能体路径规划方法①

禹鑫燚② 刘 飞 欧林林③

(浙江工业大学信息工程学院 杭州 310023)

摘 要 在多智能体路径规划问题中,每个智能体需要互相协调来完成共同的全局目标,智能体之间通常需要显式的通信策略。传统的多智能体路径规划算法受限于实时性、扩展性、不完全通信等问题,很难适用于复杂的工作环境中。为了解决多智能体工作环境中的通信问题,本文提出了一种基于图神经网络(GNN)的路径规划方法。该方法首先通过卷积神经网络(CNN)在局部观测中采集特征数据,由图神经网络在智能体之间传递这些数据。其次,为了减少智能体的惰性,提出了一种新的奖励函数,鼓励智能体更积极地探索并学习有效的协调策略。接着通过集中式收集数据训练、分布式执行提高学习效率。最后,进行多个环境下的仿真实验评估本文提出的算法,并与其他算法进行对比,验证了算法的有效性和可扩展性。

关键词 路径规划;多智能体强化学习;图神经网络(GNN);多智能体通信

近年来,随着计算机技术、机器学习算法的发展和社会的需要,移动机器人领域迎来了新的研究热潮。越来越多的移动机器人逐渐从理论走向了实际应用,如物流运输、无人驾驶、仓库智能巡检等应用,移动机器人路径规划技术的研究变得越来越重要。有效的路径规划算法能够使智能体得以高效地完成任务。

移动机器人路径规划的目标是以最低代价为一个机器人或一个机器人集群规划出一条或多条的无碰撞路径。目前,移动机器人的实际应用环境正变得越来越复杂,单个智能体执行任务往往效率过低。而多智能体系统在应用时具备鲁棒性强以及承载能力强等优点[1],部署多个智能体能够以较高效率来完成给定任务,因此采用多个智能体执行任务更具有优势。多智能体路径规划任务中的关键约束条件是机器人能够遵循规划的路径而不互相碰撞,其本身是一种典型的非确定性多项式难(non-deterministic polynomial-hard,NP-hard)问题。多智能体路径规划可以用于实际的各种行业中,如仓库管理、机场牵引、自动驾驶汽车、数字娱乐等等[2-5]。通常来说,求解多智能体路径规划问题的方法有 3 类,包括耦合、解耦合和动态耦合方法[6-7]。虽然耦合的方法能够保证解决方案的最优性和完整性,但往往扩展性会随着机器人数量的增加逐渐变差。解耦的方法为每个机器人单独规划轨迹,并只有在发生路径冲突时才重新规划路径。动态耦合方法结合了两者的优点更为完善,通常在规划期间根据需要扩大搜索空间,与耦合式方法比显著降低了计算成本,相较于解耦合方法考虑了更多的智能体交互。

随着机器学习的发展,在人工智能领域中专家学者们对多智能体路径规划的研究也越来越深入[8]。基于机器学习的方法能够将在线计算的负担卸载到离线方法中去,在解决机器人控制方面的问题时有显著的优势[9]。强化学习是机器学习的一个领域,在智能体与环境的交互过程中通过学习策略来获取最大化的回报或实现特定的目标[10]。

① 国家自然科学基金(62373329)和浙江省自然科学基金白马湖实验室联合基金(LBMHD24F030002)资助项目。
② 男,1979 年生,博士,副教授;研究方向:机器人感知、示教、控制以及人工智能在机器人中的应用;E-mail:yuxy@zjut.edu.cn。
③ 通信作者,E-mail:oulinlin@zjut.edu.cn。
(收稿日期:2023-02-24)

对于多个决策智能体联合行动影响共同环境的问题,多智能体强化学习模型成为了一种自然的解决方案,智能体不仅需要完成自己的任务,还需要学习如何与其他智能体合作,更加有效地协调整体的行动以实现全局目标。图神经网络通过将智能体系统建模成一个图来获取智能体之间的通信和协调关系。通过图神经网络,每个智能体可以获取来自附近智能体之间的信息进行信息共享。

本文提出了一种基于图神经网络(graph neural network,GNN)的多智能体路径规划方法,以多智能体强化学习为基础,通过图神经网络来实现智能体之间显式的协调策略。本文的主要贡献点如下:(1)为了应对部分可观测环境中的智能体协调问题,通过图神经网络来聚合智能体及其邻居的观测信息,来学习显式的通信策略;(2)针对稀疏奖励问题设计了一种奖励框架,可以减少机器人在到达终点后的惰性并鼓励智能体向达成全局目标前进;(3)提出了一种集中式训练、分布式执行的多智能体路径规划方法,在训练完成后可以实现模型的扩展性,在不同智能体、障碍物密度的地图上同时能够达到规划目的。

1 相关工作

1.1 传统多智能体路径规划

多智能体路径规划是一个非确定性多项式难(NP-hard)问题,由一张图和多个智能体组成,是单智能体路径规划问题的延伸[11-12]。最初的方法是将多智能体系统看作一个单独的整体,接着调用标准 A * 算法来求解问题。这样的方法需要维护巨大的搜索树形结构,属于集中式的多智能体路径规划方法。集中式算法考虑了所有相关智能体的联合配置空间,具有产生最佳和完备计划的优势,但往往计算代价较高[13],复杂度随着机器人数量的增多呈指数型上升。除了集中式的路径规划算法外,还有解耦的和动态耦合的算法,这两种算法的计算效率较高,相对集中式算法更为适用。

分布式的算法先为每个机器人规划自身的路径,接着改变这些路径来避免碰撞。因为可以在低维度空间中规划单独的路径,并对碰撞冲突进行调整,所以分布式的方法可以快速找到大型智能体系统的解决方案[14]。在调整碰撞路径时,有 2 种解决方案:基于速度的规划算法和基于优先级的规划算法。速度规划算法先确定每个机器人的路径,然后在碰撞的路径上调整速度分布从而避免碰撞[15]。特别地,最佳相互避免碰撞算法(optimal reciprocal collision avoidance,ORCA)在线调整机器人的速度大小和方向来避免碰撞。优先级规划者为每个机器人分配优先级,并按照优先级递减顺序规划各个路径。优先级低的机器人将优先级高的视为障碍物,并在与其发生冲突时主动避让或者修改自身的路径[16-18]。分布式算法的主要缺点是所使用的低维搜索空间仅代表配置空间的一小部分,这些方法是不完备的,无法为所有可解问题找到路径[19]。

近年来涌现出了一些动态耦合的多智能体路径规划算法,该类方法比解耦方法存在更多的智能体间交互。动态耦合方法所遵循的常见方法是在规划期间根据需要扩大搜索空间。典型的算法是基于冲突的搜索算法(conflict-based search,CBS)及其变体,它们为个体机器人规划并构建约束,从而在不探索高维空间的情况下找到最优或接近最优的路径[20-22]。M * 算法改造 A * 算法为解耦合算法,在 A * 的基础上,根据智能体之间的冲突动态地改变分支因子来优化路径规划过程。M * 的改进版本——递归 M * 算法(recursive M *,RM *)将路径有冲突的机器人按独立的冲突划分为机器人子集,接着在这些子集上递归地调用 M * 算法[23]。运算符分解递归 M * 算法(operator decomposition recursive M *,ODRM *)进一步改进了 M * 算法,通过将智能体集分解为独立的碰撞集并结合算子分解减少了必须进行联合规划的智能体集,在搜索过程中保持较小的分支因子[24-25]。

1.2 基于学习的多智能体路径规划

基于学习的路径规划算法有着传统算法无法比拟的优势。在集中式框架中,所有智能体的经验综合可以用于解决问题的共同方面(例如网络输出或价值、优势计算)。当集中学习网络输出时,通过共享神经网络某些层的权重进行更快更稳定的计

算[14]。集中式多智能体强化学习框架通过将所有主体的观测信息聚集到一个独立的学习过程中,在处理部分可观测系统时有显著效果。与集中式的传统多智能体路径规划算法相同,基于集中式框架的多智能体路径规划算法的效果也会随着机器人队伍的扩大而下降。而在分布式的策略学习中,每个智能体学习单独的策略并在训练中带有一定的合作性,文献[9,26-28]详细研究了这类算法。文献[14]提出了 PRIMAL(pathfinding via reinforcement and imitation multi-agent learning)多智能体路径规划框架,使用行为克隆来学习专家算法的路径规划策略,赋予团队智能体隐式的合作性,采用 A3C(asynchronous advantage actor-critic)算法来提高智能体的规划表现。文献[29]将路径规划问题扩展到长期规划问题,提出了基于多智能体强化学习得到 PRIMAL$_2$ 框架。

在实际的应用场景中,智能体之间的协作更为重要,文献[30]通过实例化 GNN 启发的结构来学习连续通信,提出了一种具备扩展性的通信方法。与本文类似的工作[31]提出了图卷积强化学习,图卷积用于适应多智能体环境中的底层图的动态关系,关系核用来捕捉智能体之间的相互作用。文献[32-34]研究表明,GNN 用于学习通信策略可以在部分可观测环境中成功地完成多智能体协调功能。文献[34]更适用于部分可观测环境,利用卷积神经网络从局部观测中提取出足够的特征,并利用图神经网络在智能体间交流这些特征,最终训练该模型以模仿专家算法。

2 基于图神经网络的多智能体路径规划

2.1 多智能体路径规划问题描述

多智能体路径规划是一个地图中的数个智能体进行有目标的移动的决策过程。本文将部分可观测环境下的多智能体路径规划问题定义成部分可观测马尔可夫决策过程,该模型将路径规划问题转化成序列决策问题。决策过程中的每一步动作构成了一个完整的路径规划序列。每个智能体需要在时间步 t 时选择动作以使得所有智能体以无碰撞路径完成各自的路径规划任务。

2.2 地图环境

考虑如图 1 所示类似仓库环境的多智能体路径规划场景 W。W 是一个二维的栅格地图,其中包含了一些静态的随机障碍物,智能体需要在地图中按照任务无碰撞地到达目标点。方块代表不同的智能体,它们各自的目标点为对应编号的圆形。栅格地图中的每个单元有 3 个状态:存在障碍物、空置和被智能体占用。为了模拟真实场景,每个智能体只能获得以自身为中心的部分地图信息。在具体实现时,该区域可以自由设置大小。在每个时间步时,每个智能体可以选取 5 个离散动作:上、下、左、右、静止不动,每次移动时智能体均移动一个单位。若移动后的位置将与其他智能体发生碰撞或与边界、障碍物发生碰撞时,智能体将不采取该动作。面对不同的地图场景,本文旨在研究一种能够动态地生成路径策略、达成多智能体的交互协作以满足不同地图场景下的多智能体路径规划需求。

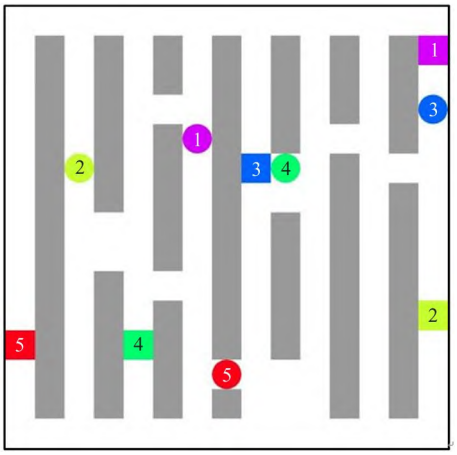
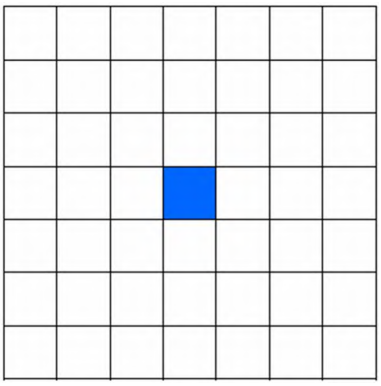


图 1 多智能体路径规划场景

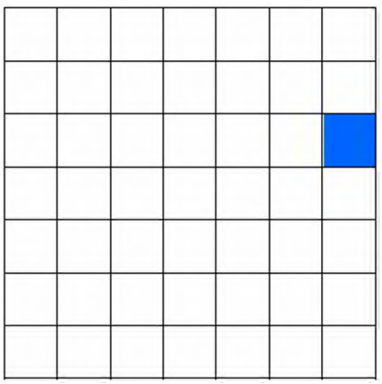
2.3 局部观测处理

在环境 W 中,每个智能体都有一个局部视野,在局部视野外智能体无法获得任何信息。设每个智能体在每个时间步 t 时可观测的局部地图信息为 z_t^i,其形状为一个以智能体所在位置为中心的正方形。若该正方形有部分落在全局地图之外,将以障碍物对该局部观测进行填充。局部地图输入到智能体自身的卷积神经网络中进行编码,使用卷积神经网络可以将输入映射处理为更高层的特征张量,便于进行特征提取。之后,卷积神经网络输出的特征

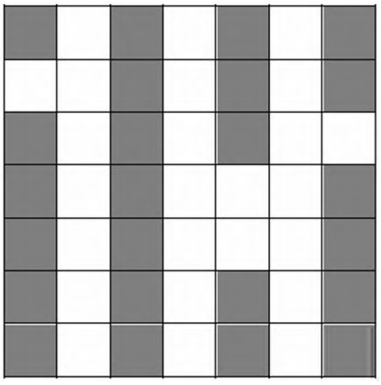
向量将由通信网络进行传递。以图 1 中的 3 号智能体为例,设局部观测大小为 7 ×7 单位栅格的正方形,对其处理如图 2 所示,可以将其分为 3 个通道。通道 1 代表所有局部观测内的智能体位置,如图 2(a)所示,可视范围内没有其他智能体。通道 2 代表中心智能体的目标点,若目标点在局部观测之外,则将该点投影在局部观测的边界处,如图 2(b)所示。通道 3 代表局部观测内的地图中的障碍物信息,如图 2(c)所示。



(a) 智能体位置



(b) 智能体目标点投影



(c) 障碍物地图

图 2　局部观测处理

2.4　奖励结构设计

在本文中,每个智能体的奖励单独计算,具体奖励设计如表 1 所示。若智能体向任意空的栅格移动且没有到达终点,则该智能体获得 -0.3 的惩罚。若智能体移动后到达终点则会获得 1.0 的奖励。若在某个时间步时,智能体选择的动作导致其与其他智能体、边界或障碍物发生碰撞,将会获得 -2.0 的较大惩罚,且该智能体会静止在原地。为了鼓励智能体探索和学习更有效的策略,智能体采取静止的动作且不在终点时则会得到一个 -0.5 的惩罚。若智能体采取静止的动作且在终点时,它将获得为0.0 的奖励。

除了碰撞惩罚外,文中还设置了一个阻塞惩罚。这种惩罚鼓励智能体离开自己的目标点而达成协作的目标,抵消智能体停留在目标点上经历的局部最大奖励。若一个智能体待在终点的同时阻碍另一个智能体到达其目标点,该智能体将收到一个较大的惩罚(实现时设置为 -1.5)。若一个智能体阻碍另一个智能体到达其目标点或使另一个智能体显著增加到目标点的步数时,则称该智能体阻塞了其他智能体,也将受到一个较大的惩罚。本文中使用标准 A * 算法来计算智能体从当前位置到终点的路径长度。首先确定环境中某智能体到其终点的路径长度,接着移除环境中其他所有的智能体,计算该智能体到达终点的路径长度。若第二条路径长度比第一条长度段 10 个栅格长度,则该智能体视野范围内的其他智能体处于阻塞状态。由于 A * 算法的计算负载较小,加入阻塞惩罚并不会影响训练效率。

表 1　奖励设计

动作	奖励
移动(上/下/左/右)	-0.3
静止(不在终点)	-0.5
静止(在终点处)	0.0
碰撞	-2.0
阻塞	-1.5
到达终点	1.0

2.5　多智能体通信

通过一张图 $\boldsymbol{G}=(\boldsymbol{V},\boldsymbol{E})$ 来为多智能体之间的

通信建模。点集 $\boldsymbol{V}=\{1,2,\cdots,N\}$ 代表每个智能体个体,边集 $\boldsymbol{E}=\boldsymbol{V}\times\boldsymbol{V}$ 代表智能体之间的通信连接。将能够与智能体 $i\in\boldsymbol{V}$ 交流的智能体集合定义为 $\boldsymbol{N}_i=\{j\in\boldsymbol{V}:(j,i)\in\boldsymbol{E}\}$。邻接矩阵 $\boldsymbol{S}$ 表示图中的连通性。若节点 j 属于 i 的邻点集,则邻接矩阵 $\boldsymbol{S}$ 中的项 $[\boldsymbol{S}]_{i,j}$ 为 1,否则为 0。

将所有智能体发送的消息集定义为 $\boldsymbol{X}$。由智能体 i 发送的信息定义为 $\boldsymbol{x}_i=[\boldsymbol{X}]_i$(矩阵 X 的第 i 行)。聚合图神经网络操作多个通信 hop k($k\in0,1,\cdots,K$),每个通信 hop k 的连通性通过将邻接矩阵提升到 k 次幂($\boldsymbol{S}^k$)来计算。对于每一个通信 hop k,通过排列不变运算来聚集消息,并根据该聚合数据来计算下一则消息。图形移位运算符 $\boldsymbol{S}$ 将消息信号 $\boldsymbol{X}$ 移位到节点上,一系列滤波器 $\boldsymbol{H}_k$ 用来聚合来自多个通信 hop 的数据:

$$[\boldsymbol{SX}]_i=\sum_{j\in\boldsymbol{N}_i}\boldsymbol{S}_{ij}\boldsymbol{X}_j \tag{1}$$

$$\boldsymbol{X}'=g_\eta(\boldsymbol{X};\boldsymbol{S})=\sum_{k=0}^{K}\boldsymbol{S}^k\boldsymbol{X}\boldsymbol{H}_k \tag{2}$$

其中,g 是以 $\eta=\{\boldsymbol{H}_k\}_{k=1}^{K}$ 为参数的函数,$\boldsymbol{X}'$接收所有智能体的数据总和。本文应用了非线性函数 σ,该过程级联了 L 次:

$$\boldsymbol{X}_l=\sigma(g_{n_l}(\boldsymbol{X}_{l-1};\boldsymbol{S}))\quad\boldsymbol{X}_0=\boldsymbol{X} \tag{3}$$

如式(1)所示,这一系列运算可以在每个智能体上本地执行。参数 η_l 在各个智能体之间共享。本文中的智能体为同质智能体,协同完成路径规划目标。在具体实现中,每个智能体只具有部分视野,并且智能体之间的通信只发生在它们之间的距离小于固定值时。

2.6 多智能体强化学习

本节提出了基于聚合图神经网络和多智能体强化学习的路径规划方法。聚合图神经网络用来达到显式的智能体间通信,通过多智能体强化学习最大化个体奖励,最终学习到一种分布式的本地路径规划策略。

2.6.1 算法框架

智能体 i 的策略 $\boldsymbol{\pi}_\theta^i$ 根据局部观测 $\boldsymbol{z}_t^i$ 和来自其他多智能体的通信消息来决定在当前的动作 a_t^i。通信消息又取决于所有智能体的局部观测 $\boldsymbol{z}_t^i$ 以及通信拓扑图 $\boldsymbol{E}_t$。集中式网络 $\boldsymbol{V}$ 需要拼接智能体的局部观测 $\boldsymbol{z}_t^i$ 以及全局地图信息 $\boldsymbol{s}_t$ 进行集中式训练,在后期测试时,仅需要每个智能体上的分布式网络进行决策。整体的算法结构如图 3 所示。

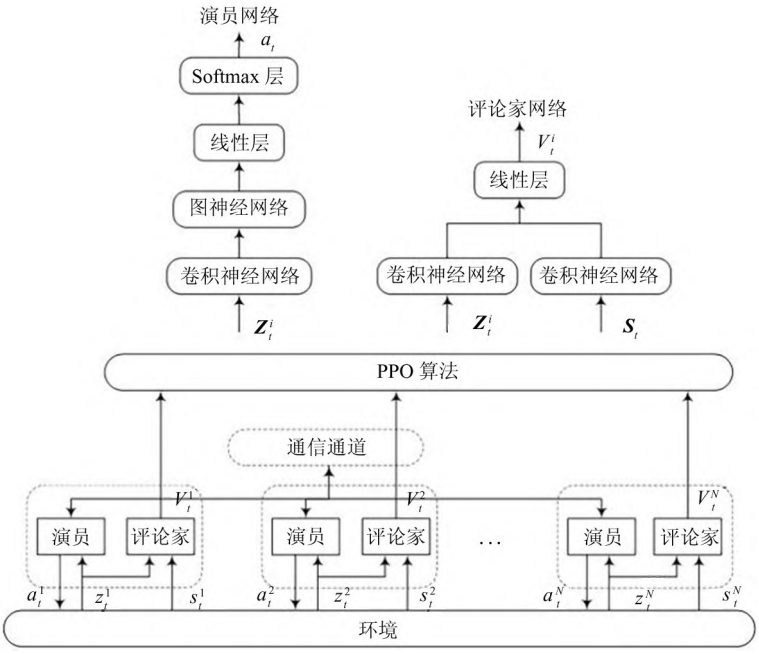


图 3 算法框架

2.6.2 网络结构

本文使用卷积神经网络(convolution neural network,CNN)编码输入信息。使用 3 层 Conv2d 网络来提取信息,接着经过 MaxPool2d 层,内核均为 3 ×

3,步长为 1。文中使用单层的卷积神经网络,将输入和输出的数量设置为 128。在每一个时间步时,策略网络都需要输出一个最优动作,策略网络由多层感知机(multi-layer perceptron,MLP)构成。每个智能体将经过 GNN 后的聚合特征输入 MLP,由于所有智能体都是同质的,因此策略网络的权重可以在所有智能体中共享。在策略网络进行输出时,本文使用 softmax 层进行解码,将含有 128 个特征的 GNN 输出转换成对应 5 个动作的概率分布。在初始训练时,智能体的动作由网络产生的随机策略给出。智能体得到的最终路径由一组动作表示。

2.6.3 合作学习

在多智能体强化学习中,每个智能体应该尽可能地在不影响其他智能体到达目标点的情况下获得最大的奖励值。

在合作学习中,每个智能体存在自身的奖励,并且系统中会有一个集中的评论家。每个智能体 i 的折扣回报为 $G_t^i = \sum_{l=0}^{\infty}\gamma^l r_{t+l}^i$。集中式值函数记作:$V^{\pi,i}(s_t) = E_t[G_t^i \mid s_t]$ 在状态 s_t 时估计每个智能体 i 的价值。对应的状态价值函数为 $Q^{\pi,i}(s_t,a_t) = E_t[G_t^i \mid s_t,a_t]$。优势函数记作:$A^i(s_t,a_t) = Q^i(s_t,a_t) - V^i(s_t)$,本文使用广义优势估计(generalized advantage estimate,GAE)来估计优势。集中值函数通过带有参数 ϕ 的网络 $\boldsymbol{V}_\phi$ 来拟合。最后,智能体 i 的策略可通过策略梯度法优化。

设每个智能体 i 在时间步 t 时的局部观测地图为 $\boldsymbol{z}_t^i$,智能体使用编码器 $f_v(\boldsymbol{z}_t^i):\boldsymbol{Z}\rightarrow R^F$(如卷积神经网络、多层感知机等)将观测值编码。编码后的观测值组成为 $\boldsymbol{X} = [f_v(\boldsymbol{z}_t^1), f_v(\boldsymbol{z}_t^2),\cdots, f_v(\boldsymbol{z}_t^N)]^{\mathrm{T}}$。每局部编码通过聚合图神经网络与邻近智能体共享。聚合的数据用于通过带有 softmax 函数的多层感知机输出动作分布。最终,本地策略以 $\theta = \{v,\eta\}$ 为参数,定义为 $\boldsymbol{\pi}_\theta^i(a_t^i \mid \boldsymbol{z}_t,\boldsymbol{E}_t)$。本文将本地策略记作 $\boldsymbol{\pi}_\theta^i(a_t^i \mid \boldsymbol{z}_t)$,省略了 E_t 的依赖。因为显式的通信策略的存在,尽管策略是分布式的且在本地执行的,每个智能体的策略也会由邻近智能体的观测值而决定。

在加入智能体之间的通信机制后,智能体的动作也会使得其他智能体的奖励上升。合作策略梯度可写作:

$$g_k^i = E_{\boldsymbol{\pi}}[\sum_{j\in V}\nabla_\theta \log \boldsymbol{\pi}_\theta^j(a^j \mid z)A^i(\boldsymbol{s})] \tag{4}$$

具体的合作学习策略梯度算法如算法 1 所示。

算法 1 基于图神经网络的合作策略梯度算法

输入:初始化策略网络参数 θ 以及用于计算优势函数的价值网络参数 ϕ

for $k = 1,2,3,\cdots$ **do**
  执行策略 $\boldsymbol{\pi}_\theta^i$,收集轨迹集 $\boldsymbol{D}_t$
  计算优势函数 $\hat{A}^i(\boldsymbol{s}_t)$
  根据 $g_k^i = \frac{1}{|\boldsymbol{D}_k|}\sum_{\tau\in\boldsymbol{D}_k}\sum_{t\in\tau}\sum_{j\in V}\nabla_\theta \log \boldsymbol{\pi}_\theta^j(a^j \mid z)A^i(\boldsymbol{s}_t)$
  估计策略梯度 $g_k = \sum_{i\in V} g_k^i$
  策略更新 $\theta \leftarrow \theta + \alpha_k g_k$
  更新价值网络
end for

3 仿真实验

设多智能体路径规划的环境为一个 $W\times H$ 的栅格地图,环境中包含了 N 个智能体 $\boldsymbol{V} = \{1,2,\cdots,N\}$。智能体 i、$j\in\boldsymbol{V}$,只有当 i、j 之间的距离小于 d 时它们才会发生通信。智能体之间的通信生成通信图 $\boldsymbol{E}_t$,表示为邻接矩阵 $\boldsymbol{S}_t$。

在具体实现中,本节使用了一张 15×15 单位栅格的正方形地图,并用 8 个同质智能体进行训练。障碍物的密度定义为障碍物所占面积与环境总面积的比值 $\rho_{obs} = \frac{n_{obs}}{(W\times H)}$。在每一幕开始时以 0.2 的障碍物密度随机初始化地图环境,实验的总时间步为 1×10^6。考虑大小为 7×7 单位栅格的局部观测,通信半径设置为 4。每一幕限制在 50 个时间步内,若所有智能体达到目标点或者超过设置的步长,则一幕结束。在每一幕结束时,记录下到达目标点的智能体数量以及智能体的路径长度,以进行效果评估。

本文使用 on-policy 的近端策略优化算法(proximal policy optimization, PPO)来整合策略梯度。

3.1 实验设置

本节的实验使用的是一台配备八核 4.2 GHz

AMD 3700x CPU 和英伟达 RTX 3070 GPU,内存分别为 16 GB 和 8 GB 的台式电脑。2.5.2 节中提到的网络由 PyTorch v1.9.4(Python 3.9)实现,由 Cuda v11.2 应用程序接口(application programming interface,API)进行计算加速。本节使用 Adam 优化器进行网络参数优化,演员(actor)和评论家(critic)的学习率均设置为 3×10^{-4}。折扣因子 γ 为 0.99,PPO 算法的裁剪参数 ε 为 0.20,GAE 的偏差-方差系数(λ)为 0.96,限制新旧策略差异的系数(entropy)为 0.01。训练中的一批次大小为 500,最小批次大小为 100,每个训练批次进行 5 次更新。

3.2 实验结果及分析

3.2.1 对比实验

在评价算法效果时,本节采用 100 张不同的地图来测试,每张地图的障碍物密度设置为 0.2 且障碍物随机生成。将训练完成的模型在智能体数量为 3~8 的地图上进行测试,实验的成功率如图 4 所示。本节将智能体的通信 hop 设置在[1,2,3]范围内进行测试,当 $k=1$ 时智能体之间不存在通信交互。因此当智能体数量变大时,模型的性能表现显著降低。当 $k=2$ 或 3 时,由于地图中的智能体数量较少,成功率的变化率较低。本节将实验结果与分布式的隐式协调多智能体路径规划算法框架 PRIMAL 进行对比,在障碍物密度为 0.2 的地图环境中测试结果与其相近,效果表现略优于 PRIMAL。此外,本节还将测试结果与 ORCA(optimal reciprocal collision avoidance)的离散版本进行了对比,成功率显著优于 ORCA。

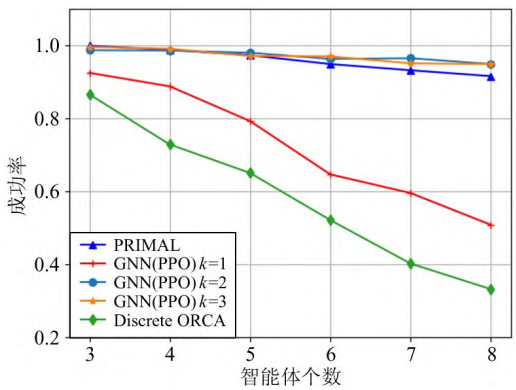


图 4 成功率对比

3.2.2 泛化分析

本小节中首先将地图扩展到 40×40 单位栅格大小,并使用更大数量的智能体编队,以测试算法的可扩展性。与上节相同,将障碍物密度设置为 0.2 且障碍物位置随机生成。本节将智能体编队规模扩展到训练的数倍,统计当智能体个数在{20,25,30,35,40,45,50}时所有智能体完成路径规划的成功率。将智能体的通信 hop 设置为 3,在 100 个不同的地图环境中进行泛化性测试,实验结果如图 5 所示。

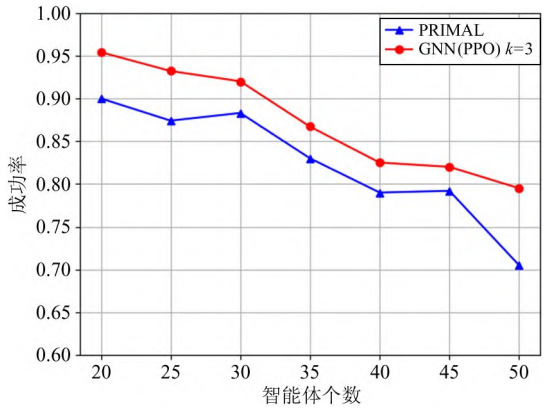


图 5 扩展环境中的路径规划成功率对比

由图可知,当智能体编队规模较小时,算法模型的表现较好,成功率较高。随着智能体数量的显著增多,算法的效果逐渐下降。得益于显式通信机制的加入,本文提出的多智能体路径规划算法的效果优于只含有隐式通信的 PRIMAL 算法。

其次,测试了在更大地图尺寸以及更多智能体数量环境下的算法成功率表现。图 6 展示了本文提出的方法在地图尺寸为80×80单位栅格、不同障碍

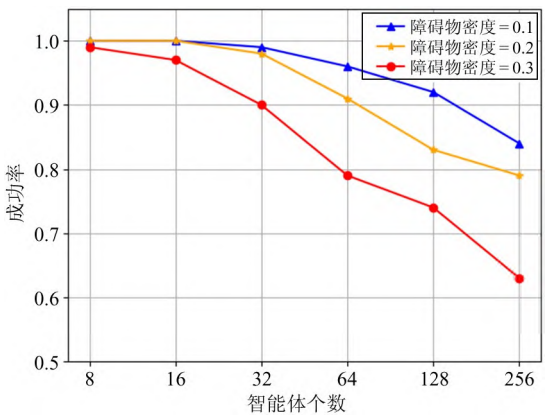


图 6 大尺寸环境中不同障碍物密度下的路径规划成功率

物密度下的路径规划成功率,障碍物密度分别设置为 0.1、0.2、0.3。由曲线图可知,当智能体数量较少时,障碍物密度对算法成功率几乎不产生影响。当智能体数量扩展到 64 时,障碍物密度为 0.3 的环境中的成功率下降明显,而障碍物密度为 0.1、0.2 的环境中成功率则小幅下降。当智能体数量扩展到 128、256 时,3 种障碍物密度环境下的成功率都显著下降,障碍物密度越小的环境中算法的成功率越高。

3.2.3　路径规划结果

以图 7 所示的密集栅格地图为例,地图尺寸为 10×10 单位栅格,障碍物密度为 0.4,环境中包含 5 个智能体,它们分别位于不同编号的方块点处,各自的目标点为对应编号的圆形。当智能体到达目标点后,目标点就会变成黑色(图中未展示出智能体的每一步移动)。

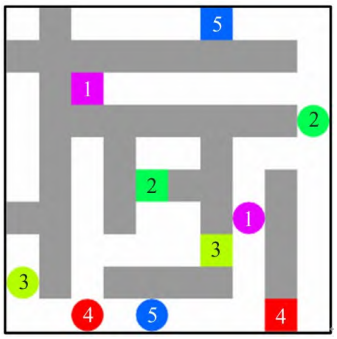
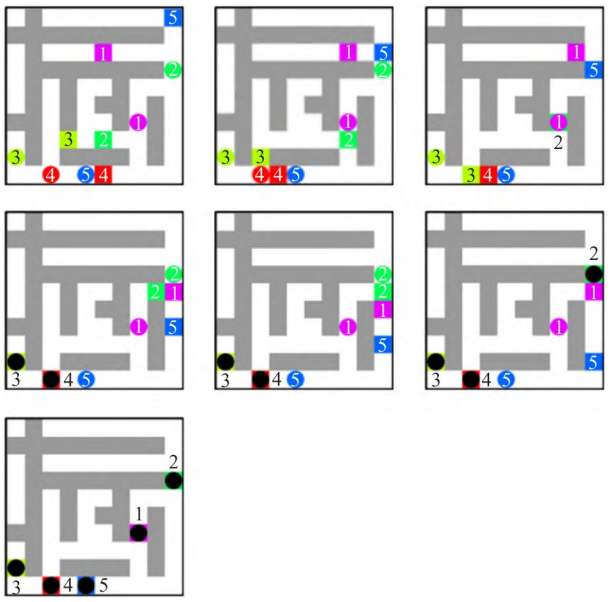
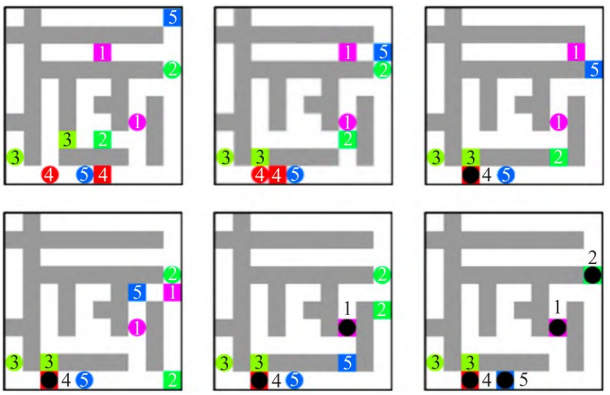


图 7　初始地图

本文提出算法的路径结果如图 8(a)所示,5 号智能体的路径不会受到其他智能体的影响,而 3 号和 4 号智能体在进行第 6 步动作时若继续按原方向移动将发生点冲突。在此时间步时,4 号智能体会采取静止等待策略,在 3 号智能体通过后再移动。在时间步 9 时,2 号智能体和 1 号智能体将会发生路径冲突,此时,1 号智能体会执行让步策略向下移动一步,协调以达到全局目标。图 8(b)展示了 PRIMAL 算法在相同地图上的路径规划结果。相较于本文提出的算法,2 号智能体到达目标点的路径长度明显增加。由于没有通信机制的加入,在 3 号和 4 号智能体发生点冲突时,4 号智能体优先采取动作移动至目标点,3 号智能体将其视作障碍物,采取静止等待动作,最终无法协作到达目标点。



(a) 本文提出算法的路径规划结果



(b) PRIMAL 算法的路径规划结果

图 8　路径规划结果

4　结　论

本文研究了局部可观测环境下多智能体的路径规划方法,提出了一种基于图神经网络的多智能体路径规划方法,能够使智能体之间形成通信并在无法获取全局信息的情况下生成无碰撞的路径。将智能体的局部观测分为 3 部分,由卷积神经网络提取特征,接着由图神经网络在智能体之间传递这些特征信息,使得系统能够了解哪些信息与智能体编队相关。此外,为了解决稀疏奖励问题,文中提出了一种多步补偿的奖励结构。本文使用了中心式训练分布式执行的强化学习框架,以一个集中式的价值网络来评价智能体的策略对整体的影响,每个智能体的策略由自身的动作网络输出,这些网络的权重是

共享的。

实验结果表明,文中提出的方法可以在部分可观测环境中实现智能体之间的高效通信。训练完成的模型可以扩展到含有不同个数智能体的不同环境中使用,通过权重共享和分布式执行可以降低计算的负载,保证计算上的可行性。

参考文献

[1] 孟慧婕. 静态环境下多移动机器人路径规划方法的研究[D]. 天津: 河北工业大学, 2016.

[2] WURMAN P R, D'ANDREA R, MOUNTZ M. Coordinating hundreds of cooperative, autonomous vehicles in warehouses[J]. AI Magazine, 2008,29(1):1752-1759.

[3] MORRIS R, PASAREANU C S, LUCKOW K, et al. Planning, scheduling and monitoring for airport surface operations[C]//Workshops at the 30th AAAI Conference on Artificial Intelligence. Phoenix, USA: AAAI Press, 2016:608-614.

[4] VELOSO M, BISWAS J, COLTIN B, et al. Cobots: robust symbiotic autonomous mobile service robots[C]// The 24th International Joint Conference on Artificial Intelligence. Buenos Aires, Argentina: Morgan Kaufmann, 2015:4423-4429.

[5] MA H, YANG J, COHEN L, et al. Feasibility study: moving non-homogeneous teams in congested video game environments[C]//The 13th Artificial Intelligence and Interactive Digital Entertainment Conference. Salt Lake City, USA: AAAI Press, 2017:1-4.

[6] LEROY S, LAUMOND J P, SIMÉON T. Multiple path coordination for mobile robots: a geometric algorithm[C]//The 16th International Joint Conference on Artificial Intelligence. Stockholm, Sweden: Morgan Kaufmann, 1999:1118-1123.

[7] BERG J, GUY S J, LIN M, et al. Reciprocal n-body collision avoidance[C]//The 14th International Symposium on Robotics Research. Berlin Heidelberg: Springer, 2011:3-19.

[8] STERN R. Multi-agent path finding——an overview[J]. Artificial Intelligence, 2019,2019:96-115.

[9] GUPTA J K, EGOROV M, KOCHENDERFER M. Cooperative multi-agent control using deep reinforcement learning[C]//International Conference on Autonomous Agents and Multi-Agent Systems. Sao Paulo, Brazil: Springer, 2017:66-83.

[10] 邱锡鹏. 神经网络与深度学习[M]. 北京:机械工业出版社, 2020:330-331.

[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. Honolulu, USA: AAAI Press, 2019:7643-7650.

[12] LAVALLE S M. Planning algorithms[M]. Cambridge: Cambridge University Press, 2006:43-47.

[13] 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. Las Vegas, USA: IEEE, 2020: 11785-11792.

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

[15] CUI R, GAO B, GUO J. Pareto-optimal coordination of multiple robots with safety guarantees[J]. Autonomous Robots, 2012,32(3):189-205.

[16] MA H, TOVEY C, SHARON G, et al. Multi-agent path finding with payload transfers and the package-exchange robot-routing problem[C]//Proceedings of the AAAI Conference on Artificial Intelligence. Phoenix, USA: AAAI Press, 2016:3166-3173.

[17] CÁP M, NOVÁK P, SELECKY M, et al. Asynchronous decentralized prioritized planning for coordination in multi-robot system[C]//2013 IEEE/RSJ International Conference on Intelligent Robots and Systems. Tokyo, Japan: IEEE, 2013:3822-3829.

[18] ERDMANN M, LOZANO-PEREZ T. On multiple moving objects[J]. Algorithmica, 1987,2(1):477-521.

[19] SANCHEZ G, LATOMBE J C. Using a PRM planner to compare centralized and decoupled planning for multi-robot systems[C]//Proceedings of 2002 IEEE International Conference on Robotics and Automation. Washington, USA: IEEE, 2002,2:2112-2119.

[20] BARER M, SHARON G, STERN R, et al. Suboptimal variants of the conflict-based search algorithm for the multi-agent path finding problem[C]//The 7th Annual Symposium on Combinatorial Search. Prague, The Czech Republic: AAAI Press, 2014:961-962.

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

[22] BOYARSKI E, FELNER A, STERN R, et al. ICBS: improved conflict-based search algorithm for multi-agent path finding[C]//The 24th International Joint Conference on Artificial Intelligence. Buenos Aires, Argentina: Morgan Kaufmann, 2015:740-746.

[23] WAGNER G, CHOSET H. M*: a complete multirobot path planning algorithm with performance bounds[C]// 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems. San Francisco, USA: IEEE, 2011: 3260-3267.

[24] FERNER C, WAGNER G, CHOSET H. ODRM* optimal multirobot path planning in low dimensional search spaces[C]//2013 IEEE International Conference on Robotics and Automation. Karlsruhe, Germany: IEEE, 2013:3854-3859.

[25] STANDLEY T. Finding optimal solutions to cooperative path finding problems[C]// Proceedings of the AAAI Conference on Artificial Intelligence. Atlanta, USA: AAAI Press, 2010,24(1):173-178.

[26] LOWE R, WU Y I, TAMAR A, et al. Multi-agent actor-critic for mixed cooperative-competitive environments[C]// Proceedings of the 31st International Conference on Neural Information Processing Systems. Long Beach, USA: Curran Associates Inc., 2017:6382-6393.

[27] FOERSTER J, FARQUHAR G, AFOURAS T, et al. Counterfactual multi-agent policy gradients[C]// Proceedings of the AAAI Conference on Artificial Intelligence. New Orleans, USA: AAAI Press, 2018:1-10.

[28] FOERSTER J, ASSAEL I A, DE FREITAS N, et al. Learning to communicate with deep multi-agent reinforcement learning[C]// Proceedings of the 30th International Conference on Neural Information Processing Systems. Barcelona, Spain: Curran Associates Inc., 2016:2145-2153.

[29] DAMANI M, LUO Z, WENZEL E, et al. PRIMAL$_2$: pathfinding via reinforcement and imitation multi-agent learning-lifelong[J]. IEEE Robotics and Automation Letters, 2021,6(2):2666-2673.

[30] SUKHBAATAR S, FERGUS R. Learning multi-agent communication with back propagation[C]// Proceedings of the 30th International Conference on Neural Information Processing Systems. Barcelona, Spain: Curran Associates Inc., 2016:2252-2260.

[31] TOLSTAYA E, GAMA F, PAULOS J, et al. Learning decentralized controllers for robot swarms with graph neural networks[C]//Conference on Robot Learning. Auckland, New Zealand: PMLR, 2020:671-682.

[32] 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. Las Vegas, USA: IEEE, 2020: 11785-11792.

[33] KHAN A, TOLSTAYA E, RIBEIRO A, et al. Graph policy gradients for large scale robot control[C]//Conference on robot learning. Las Vegas, USA: PMLR, 2020: 823-834.

[34] JIANG J, DUN C, HUANG T, et al. Graph convolutional reinforcement learning[EB/OL]. (2018-10-22)[2020-02-11]. http://arxiv.org/pdf/1810.09202.

A multi-agent path planning method based on graph neural network

YU Xinyi, LIU Fei, OU Linlin

(College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023)

Abstract

For the multi-agent path planning problem, each agent needs to coordinate with each other to accomplish a global goal. Explicit communication strategies are usually required between the agents. The traditional multi-agent path planning algorithms are limited by its insufficiency of instantaneity, scalability, and incomplete communication, which are difficult to be applied in complex environments. In order to solve the communication problem in multi-agent working environment, a path planning method based on graph neural network(GNN) is proposed. The method collects feature data in local observation by convolutional neural network (CNN), and GNN transmits these data among the agents. Second, to reduce the inertia of the agents, a new reward structure is proposed to encourage the agents to explore and learn effective coordination strategies more actively. Then, the learning efficiency is improved by centralized data collection for training and distributed execution. Finally, simulation experiments in different environments are conducted to evaluate the algorithm proposed in this paper and compare it with other algorithms to verify its effectiveness and scalability.

Key words: path planning, multi-agent reinforcement learning, graph neural network(GNN), multi-agent communication