多智能体专家型策略梯度的目标跟踪与清障

孙辉辉[1,2], 胡春鹤[1†], 张军国[1]

(1. 北京林业大学 工学院, 北京 100083;

2. 华北科技学院 机电工程学院, 河北 廊坊 065201)

摘要: 为适应复杂环境下目标跟踪机器人高效运动规划需求, 本文提出一种基于多智能体强化学习的专家型策略梯度(ML-DDPG)方法. 为此首先构建了基于最小化任务单元的分布式多Actor-Critic网络架构; 随后针对机器人主动障碍清除和目标跟踪任务建立了强化学习运动学模型和视觉样本预处理机制, 由此提出一种专家型策略引导的最优目标价值估计方法; 进一步通过并行化训练与集中式经验共享, 提升了算法的训练效率; 最后在不同任务环境下测试了ML-DDPG算法的目标跟踪与清障性能表现, 和其它算法对比验证了其在陌生环境中良好的迁移与泛化能力.

关键词: 移动机器人; 多智能体; 强化学习; 运动规划; 专家策略

引用格式: 孙辉辉, 胡春鹤, 张军国. 多智能体专家型策略梯度的目标跟踪与清障. 控制理论与应用, 2022, 39(10): 1854 – 1864

DOI: 10.7641/CTA.2022.10935

Target tracking and obstacle clearing with multi-agent expert strategy gradient

SUN Hui-hui[1,2], HU Chun-he[1†], ZHANG Jun-guo[1]

(1. School of Technology, Beijing Forestry University, Beijing 100083, China;

2. School of Mechanical and Electrical Engineering, North China Institute of Science and Technology, Langfang Heibei 065201, China)

Abstract: In order to satisfy the requirements of efficient motion planning for target tracking robot in complex environment, a novel multi-agent deep deterministic strategy gradient (ML-DDPG) approach is proposed based on expert knowledge. Firstly, the approach constructs a distributed multi-Actor-Critic network architecture aiming at minimizing task units, and the Markov reinforcement learning kinematic model is also established for active obstacle clearing and target tracking tasks of mobile robot. Then, the visual sample preprocessing mechanism is constructed by utilizing multilayer convolutional neural network, and an optimal target value estimation method is put forward by expert strategy guiding mechanism. Based on these innovative improvements, the training efficiency of the ML-DDPG is improved through parallel training and centralized experience sharing principle. Finally, the performance indexes for obstacle clearing and target tracking are verified in different task environments based on physical simulator. Compared with the state-of-the-art motion planning methods, the ML-DDPG performs better migration and generalization ability in unknown environments.

Key words: mobile robot; multi-agent; reinforcement learning; motion planning; expert strategy

Citation: SUN Huihui, HU Chunhe, ZHANG Junguo. Target tracking and obstacle clearing with multi-agent expert strategy gradient. *Control Theory & Applications*, 2022, 39(10): 1854 – 1864

1 引言

当今移动机器人技术飞速发展, 在各个崭新领域的应用层出不穷, 从工业生产到日常生活、从军事应用到危险探索, 无不充斥着人类活动的每个角落[1–2]. 随着作业环境复杂度提升和任务精度要求的提高, 各行业对移动机器人智能化需求也在日益增长, 力求移动机器人具备多样化任务的执行能力和更强的自主决策与持续学习技能[3]. 这将对移动机器人的运动路径规划策略提出巨大地挑战. 经典的运动规划算法如栅格法、可视图法、人工势场法、矢量场法等大多需

收稿日期: 2021−09−30; 录用日期: 2022−06−23.

†通信作者. E-mail: huchunhe@bjfu.edu.cn.

本文责任编委: 刘帅.

国家自然科学基金项目(61703047), 河北省高等学校科学技术研究项目(QN2021312)资助.

Supported by the National Natural Science Foundation of China (61703047) and the funded by Science and Technology Project of Hebei Education Department (QN2021312).

要精确完整的地图模型或者先验知识[4–6], 一些智能算法如模糊逻辑法、遗传算法和神经网络算法等需要复杂的编程[7–9], 而且环境感知与运动控制被分成了两个独立的阶段, 前一阶段的感知误差会随之累计到后面的控制模块, 进一步增加系统误差. 除此之外, 这类路径规划算法缺少自主探索和策略升级能力, 无法根据当前状态选择最适合的动作. 尤其面对跟踪、避障和清障等复杂的任务, 这些算法的精确性和泛化能力将受到大幅度的限制. 随着机器学习技术的快速发展, 深度强化学习 (deep reinforcement learning, DRL) 方法为复杂化境中的移动机器人运动控制问题带了新的解决方案[10]. 该类方法是一种集合了感知与决策为一体的自主学习方案[11], 可以降低开发人员的编程难度和对系统模型的依赖, 仅需智能体与环境之间反复地“试错”交互, 并以获取系统最大奖励为目标, 就可以直接从感知信息中学习到良好的控制策略[12]. DRL运动规划方法降低了中间环节的累计误差, 实现了从原始输入到动作输出的端对端直接控制, 尤其适用与解决机器人的避障、导航、障碍清理等任务[13].

基于深度强化学习的运动规划方法由机器人感知部件直接采集数据信息, 并通过策略网络生成机器人控制动作指令. 机器人感知部件主要以视觉传感器、激光雷达传感器和超声波传感器为主. 文献[14–16]以视觉传感器作为状态信息采集单元, 实现了利用DRL算法在仿真环境种多种动静态障碍物下的自主导航训练. 但是这些算法将高维图像数据简单处理后就直接送入强化学习网络训练, 状态维度过高, 给强化学习策略–评价网络带来压力. 文献[17–18]为了解决传统运动规划需要不断建立和更新地图信息的劣势, 使用激光雷达和其他距离传感器作为感知单元实时获取周围环境障碍信息, 采用SAC和DDPG等强化学习实现了在非结构未知环境中以及拥挤的社会环境中端到端的实时避障与导航[19–20], 但是这些方法存在网络更新步长难以选择, 动作价值估计不准, 新旧策略差异过大等情况. 因此在跟踪移动机器人的任务环境中, 具备自主跟踪、实时决策能力, 能够适用于复杂多变任务场景的机器人显得变得重要[21–22]. 文献[23] 基于双目视觉设计了主从式果园作业车辆自主跟随系统, 实现野外作业环境中的动态目标跟随.文献[24] 基于DDPG强化学习算法, 改进了经验样本池的抽取和存储方式, 结合最小安全距离模型, 设计了多目标车辆跟随决策系统. 但是这些目标跟踪任务没有考虑避障与跟踪的协调, 跟踪机器人需要频繁的改变自身状态, 从而导致跟踪目标的丢失率的增加和系统能量损耗的加剧.

面对常规运动控制方法的不足以及目前单智能体强化学习所存在的缺陷, 本文将提出一种基于多智能体专家型深度确定性策略梯度的强化学习算法, 以满足复杂环境下清障型机器人自主目标跟踪任务的需求. 在环境感知方面, 以视觉传感器作为输入, 应用卷积神经网络进行特征提取, 将降维后的特征信息输入强化学习系统中为系统决策提供数据支撑; 针对清障类型的移动机器人主体, 建立基于多智能体的专家型确定策略梯度网络结构(multi-agent deep deterministic strategy gradient policy, ML-DDPG), 利用并行化的训练方式和专家型的目标策略引导, 指导机器人在训练前期以更快的速度向着最优动作策略的方向探索, 实现模型的快速收敛与优化. 同时, 以集中式经验池共享单个智能体之间的状态信息, 提高多智能体之间的协调性与不同任务场景之间的迁移能力.

2　机器人物理平台建模

本文所述跟踪机器人系统以履带式移动机构为基础平台, 辅助以视觉传感器作为状态采集模块. 平台装载多自由度机械臂以实时清理运动过程中所遇到的小型障碍物. 跟踪机器人系统运动简图如图1所示.

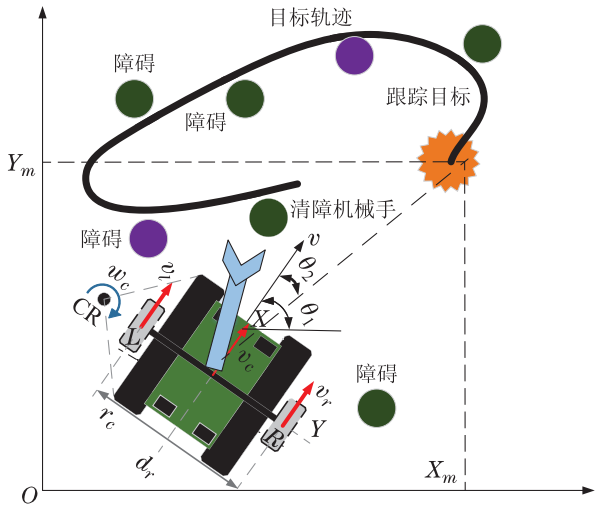


图 1　跟踪机器人运动简图

Fig. 1　Modeling sketch of tracking robot

系统包括清障型机器人本体、跟踪目标和多种动静态障碍物. 机器人的纵向速度为v, 与X轴方向的夹角为θ_1, 与目标的夹角为θ_2. 清障型机器人地盘为双履带差速机构, X为横轴线, 以Y为纵轴线. 虚拟左右轮的位置分别位于点L和R, L和R的距离为d_r. 机器人转向的圆心位于CR处, 转向角速度为w_c, X方向的线速度v_c. 根据双履带的差动速度, 可以得到机器人机身运动参数

$$\begin{bmatrix} v_c \\ w_c \end{bmatrix} = \begin{bmatrix} \dfrac{v_r + v_l}{2} \\ \dfrac{v_r - v_l}{d_r} \end{bmatrix} = \begin{bmatrix} 1/2 & 1/2 \\ 1/d_r & -1/d_r \end{bmatrix} \begin{bmatrix} v_r \\ v_l \end{bmatrix}, \quad (1)$$

其中: d_r表示虚拟轮间距, v_l和v_r分别表示虚拟左右轮的线速度. 机器人的线速度为v_c, 角速度为w_c.

机器人几何质心位置为(x_c, y_c), θ_1为航向角. 因此, 跟踪机器人的运动学方程为

$$\begin{cases} \dot{x}_{\rm c} = v_{\rm c}\cos\theta_1, \\ \dot{y}_{\rm c} = v_{\rm c}\sin\theta_1, \\ \dot{\theta}_1 = \omega_{\rm c}. \end{cases} \tag{2}$$

机器人跟踪目标的位置坐标为$(x_{\rm m}, y_{\rm m})$. 跟踪机器人与跟踪目标的实际距离为d, 最佳距离为d_0. 则综合跟踪误差为

$$\begin{cases} e_d = |d - d_0|, \\ e_\theta = |\theta_2| = |\theta_1 - a\tan\dfrac{y_{\rm c} - y_{\rm m}}{x_{\rm c} - x_{\rm m}}|, \end{cases} \tag{3}$$

其中: e_d为直线偏差, e_θ为角度偏差, d_0为场景预先设定值, $d = \sqrt{(x_{\rm c} - x_{\rm m})^2 - (y_{\rm c} - y_{\rm m})^2}$.

清障机械臂的控制方式为末端执行器的位置控制. 由深度强化学习策略输出末端执行器的广义速度, 指引末端执行器朝着待清理障碍的坐标移动. 清障机械臂为5自由度关节型机械臂, 由5个旋转关节(A1-A5)和相应的连杆机构串联组成. 根据机械臂关节结构参数, 建立D-H参数表, 如表1所示.

表 1 跟踪机器人D-H参数表

Table 1 D-H parameter table of tracking robot manipulator

i	$d(i-1)$	$\theta(i-1)$	a_i	$\alpha(i-1)$
1	L_1	θ_1	0	0
2	0	θ_2	L_1	−90
3	0	θ_3	L_2	0
4	0	θ_4	L_3	0
5	L_5	θ_5	0	−90

以机器人本体坐标系为基座坐标系, 清障机构为末端坐标系, 对于五自由度机械臂建立坐标变换[25]

$$^0_5A = {}^0_1A^1_2A^2_3A^3_4A^4_5A, \tag{4}$$

其中$^{i-1}_iA$表示当前坐标相对于前一坐标的转换矩阵.

根据表1中的D-H参数, 可以得到相邻关节的位置变换矩阵

$$^0_5A = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix},\ ^0_1A = \begin{bmatrix} c\theta_1 & 0 & s\theta_1 & 0 \\ s\theta_1 & 0 & -c\theta_1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$^1_2A = \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & L_2c\theta_2 \\ s\theta_2 & s\theta_2 & 0 & L_2s\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},\ ^2_3A = \begin{bmatrix} c\theta_3 & -s\theta_3 & 0 & L_3c\theta_3 \\ s\theta_3 & c\theta_3 & 0 & L_3s\theta_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$^3_4A = \begin{bmatrix} c\theta_4 & -s\theta_4 & 0 & L_4c\theta_4 \\ s\theta_4 & s\theta_4 & 0 & L_4s\theta_4 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},\ ^4_5A = \begin{bmatrix} c\theta_5 & 0 & s\theta_5 & L_5c\theta_5 \\ s\theta_5 & 0 & -c\theta_5 & L_5s\theta_5 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{5}$$

其中: $[n, o, a]$为末端执行器姿态参数, $[p_x, p_y, p_z]$为末端执行器位置参数.

然后, 对机械臂的末端位置坐标$[p_x, p_y, p_z]$对关节向量空间求导可以得到末端执行器线速度$[v_x, v_y, v_z]$

$$\begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} = \begin{bmatrix} \dot{p}_x \\ \dot{p}_y \\ \dot{p}_z \end{bmatrix} = J_v(q)\dot{q}(\dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3, \dot{\theta}_4, \dot{\theta}_5), \tag{6}$$

其中: J_v为末端执行器的线速度雅可比矩阵, q表示关节向量空间角位移信息, $\dot{q}$为其导数, 同时代表了机械臂各关节的角速度信息.

最后, 在获得末端执行器的线速度$[v_x, v_y, v_z]$的条件下, 即可求得机械臂各关节的角速度信息

$$[\omega_1\ \omega_2\ \omega_3\ \omega_4\ \omega_5]^{\rm T} = J_v^{-1}[v_x\ v_y\ v_z]^{\rm T}. \tag{7}$$

其中J_v^{-1}为线速度雅可比矩阵的广义逆矩阵.

综上, 根据ML-DDPG所输出的末端执行器的速度信息, 即可基于式(7)求得机器人各关节角速度运动参数, 进而控制机械臂完成下一步动作.

3 多智能体确定性策略梯度

3.1 强化学习网络构建

应用多智能体强化学习架构来解决复杂环境下的清障型机器人的目标跟踪任务, 可使机器人具有更强的自主学习和决策能力, 克服单一智能体数据使用效率不高, 策略输出之间的相互干预, 难以从奖励函数中学习到有效的策略的问题.

首先需要基于马尔科夫过程建立跟踪移动机器人马尔科夫决策五元组$[S, A, P, R, \gamma]$模型. 其中$(s_1, s_2, s_3) \in S$是所有观察状态的集合, 包括目标状态, 障碍物状态和自身状态. $(a_1, a_2, a_3) \in A$表示机器人在t时刻可执行动作的集合, 包括机器人航速a_1、机器人航向角速度a_2、机械臂末端动作a_3. $R: S \times A \to R$是奖励函数, 在t时刻机器人执行的动作A后环境所给予的反馈. $\gamma \in (0, 1)$为折扣因子, 用来计算回合累计奖赏. P为状态转移概率$P|S_t \times A \times S_{t+1} \to (0, 1)$, 表示当前状态行为下, 执行动作$A$, 转移到下一状态的概率分布

$$\sum P(s_t, a_t, s_{t+1}) = 1,\ \forall_s \in S,\ \forall_a \in A, \tag{8}$$

其中: S_{t+1}表示下一时刻可能的状态, S_t表示t时刻的状态, A为t时刻机器人所执行的动作.

跟踪机器人在与环境交互中的目标是获得系统的累计回报最大化. 对于目标任务场景, 定义累计回报为

$$G_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}. \tag{9}$$

跟踪机器人的动作选择需要策略函数$\pi(s)$来指导, 策略函数是将状态映射到行为的联系. 为此建立策略函数评价指标–状态行为值函数

$$Q^{\pi}(s,a)=E_{\pi}(\sum_{k=0}^{\infty}\gamma^{k}r_{t+k+1}). \tag{10}$$

状态行为值函数$Q^{\pi}(s,a)$即是根据策略函数$\pi(s)$从状态s开始采取行为A所获得的期望回报.

本文基于深度确定性策略梯度[26–27], 改进常规的单Actor-Critic 结构, 摒弃了单个智能体独自负责多个动作输出的传统模式, 以单元化任务为目标, 构建了并行化的多智能体深度确定性网络结构. 具有多任务型的跟踪机器人可以将自身任务分解为多个单元任务, 每个智能体负责一个单元任务的学习与训练, 智能体之间在共享感知信息的同时具有独立决策并行化运行的特征. 机器人通过一次性交互的集中式学习, 即可实现实现多个策略网络同时更新和分布式的应用. 降低了单个智能体网络的复杂度, 提升网络训练的目标针对性. 利用经验池信息共享机制, 使策略网络更容易从当前奖励获取最优策略, 同时增强网络模型复杂环境中迁移的鲁棒性. ML-DDPG强化学习网络结构如图2所示.

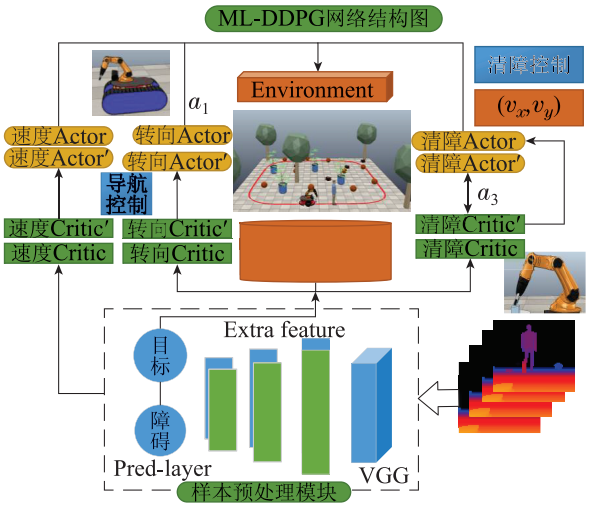


图 2　多智能体网络结构

Fig. 2　Multi-agent network architecture

ML-DDPG框架中共含有3个智能体, 每个智能体拥有自己的Actor策略网络的参数为θ和Critic价值网络的参数为w. 策略网络负责机器人的下一步动作的选择. 其中, 速度Actor1和转向Acotr2分别控制机器人的前进速度和航向角速度, 清障Actor3 负责控制机器人机械臂的末端执行器的移动速度和方向. 评价网络Critic负责评估对应策略网络的当前价值函数和目标价值.

ML-DDPG采用经验池样本回放的方式去更新Actor和Critic网络. 回放池中的样本$[S,A,S',R]$同时包括3个智能体的状态和动作信息, 单个智能体在获取到自身状态信息同时能够共享到其他智能体状态信息.

每个策略网络分为当前策略网络Actor以及目标策略网络Acotor′. 第i个Actor$_i$负责根据当前状态S选择当前动作a_i, 并和环境交互生成下一状态S'以及奖励r_i. ML-DDPG 以智能体的的累积期望奖励作为Actor$_i$更新的目标函数

$$J(\theta_i)=E_{s\sim\rho^{\pi},a_i\sim\pi_{\theta_i}}(\sum_{t=0}^{\infty}\gamma^{t}r_{it}), \tag{11}$$

其中: π_{θ_i}为当前Actor$_i$的动作选择策略, γ为折扣因子.

然后应用求解目标函数策略梯度的方式以更新策略网络参数[26–27]θ_i

$$\begin{aligned}\nabla_{\theta_i}J(\pi_i)=E_{x,a\sim D}[&\nabla_{\theta_i\pi_i}(a_i|s_i)\\&\nabla_{ai}Q_i^{\pi}(S,a_i)],\end{aligned} \tag{12}$$

其中: s_i表示第i个智能体的状态, $S=[s_1,\cdots,s_n]$表示系统状态向量, $Q_i^{\pi}(S,a_i)$为关于系统状态和当前动作的价值函数.

ML-DDPG中的价值网络同样分为当前价值网络Critic以及目标价值网络Critic′. Critic负责价值网络参数$[w_1,w_2,w_3]$的更新, 并且计算当前网络动作价值$Q(S,A,w)$. 目标价值网络Critic′负责估计目标动作价值$Q'(S',A',w')$. 价值网络Critic利用时间差分(TD)误差作为网络的损失函数

$$L(\theta_i)=E_{S,a_i,r_i,S'}[(y_i-Q_i^{\pi_i}(S,A,w_i))^2], \tag{13}$$

其中: $y_i=r_i+\gamma Q_i^{\pi_i'}(S',A',w_i')\,|_{a_i-\pi(s)}$, w_i代表的是当前价值网络参数, w_i'代表的是目标价值网络参数.

最后, 目标网络Critic′和Actor′参数的更新方式采用延迟的软更新方式, 每次复制部分当前网络参数, 防止动作价值被过高估计

$$\begin{cases}w_i'\leftarrow\tau w_i+(1-\tau)w_i',\\ \theta_i'\leftarrow\tau\theta_i+(1-\tau)\theta_i'.\end{cases} \tag{14}$$

3.2　状态样本信息预处理

深度强化学习可分为感知和决策两个部分, 感知模块将传感器获得的环境状态信息转化为更容易识别的状态表征; 决策模块根据当前状态与系统奖励值不断更新智能体策略模型; 为了减轻决策模块的压力, 本文构建新型感知模块结构, 在常规全连接层的基础上, 增加多层卷积神经网络构建特征编码器, 利用预先训练的方式, 对图像样本信息进行预处理, 提取出跟踪目标和障碍目标的特征信息, 然后送入决策模块, 感知模块中的样本预处理结构如图2中所示. 其中, 视觉传感器获取的图像数据是连续的视频流, 算法取连续的四帧作为一个状态观察, 并利用感知模块进行特征提取. 图像预处理模块3个部分: VGG-16基础网络、额外增加的卷积层Extra-Layer和和预测层Pred-Layer. 状态样本预处理模块通过VGG-base和Extra-layer网络层进行目标特征提取, 在不同尺度的特征特征图上采用卷积核来预测目标物体的类别、坐标偏移

等特征信息. 在与环境交互之前, 样本预处理模块卷积神经网络被预先训练, 保证模块在训练完成后能够正确检测是目标和障碍的特征信息. 网络预训练数据集采用COCO数据集, 预训练损失函数定义为位置误差 (locatization loss, loc) 与置信度误差 (confidence loss, conf)的加权和[28]

$$L(x,c,l,g)=\frac{1}{N}\left(L_{\text{conf}}(x,c)+\alpha L_{\text{loc}}(x,l,g)\right). \tag{15}$$

其中: N是先验框的正样本数量, x为一个指示参数, 表示先验框与真实目标匹配程度, c为类别置信度预测值, l为先验框的所对应边界框的位置预测值, g是真实目标的位置参数.

对于位置误差, 其采用Smooth L_1 loss形式; 置信度误差, 采用softmax loss函数. 跟踪机器人在利用样本预处理模块获得两类信息. 一是各个类别的置信度或者评分, 根据具体的跟踪对象, 选择正确的跟踪目标, 并且判断障碍物类别和位置; 二是边界框的位置, 包含4个值(c_x, c_y, w, h), 分别表示边界框的中坐标以及宽高.

4 专家型策略引导机制

为了缩短多智能深度确定性策略梯度的模型训练周期, 提高模型的收敛速度, 降低移动机器人前期的无效探索周期, 本文提出的ML-DDPG算法将模型训练方式分为双阶段进行. 训练第1 阶段, 基于深度确定性策略梯度网络结构, 在Actor策略网络部分融入专家经验指导机制,具体算法结构如图3所示. Actor当前网络负责选择当前动作与环境互动, Actor′目标网络由带有误差控制的PID方法代替. PID控制策略根据当前运动误差, 对机器人的下一步动作进行专家型预测, 获取接近于当前状态的最优动作, 减少网络初始阶段的随机探索与试错回合. 然后将专家优选动作A'送入Critic′目标评价网络计算目标动作价值$Q(S',A')$, 防止动作价值被过分错误高估. 训练第2阶段, 带有专家指导机制的Actor′模块将被重新替换为与Actor策略网络结构一致的目标网络, 增加网络的随机探索机制, 继续提升策略网络性能.

如图3所示, 当前策略Actor负责选择当前动作并且与环境进行互动

$$a_t=\mu(s_t|\theta^u)+N_t, \tag{16}$$

其中N_t为随机噪声.

在训练第1阶段, 具有专家指导机制的动作选择策略为

$$a'=\mu^{\text{PID}}(s')=K_{\text{P}}(e^{(t)}+\frac{1}{T_t}\int_0^t e(t)\text{d}t+T_{\text{D}}\frac{\text{d}e(t)}{\text{d}t}), \tag{17}$$

其中: $e(t)$移动机器人当前运动的状态误差, K_{p}, T_t, T_{D}为PID调节参数.

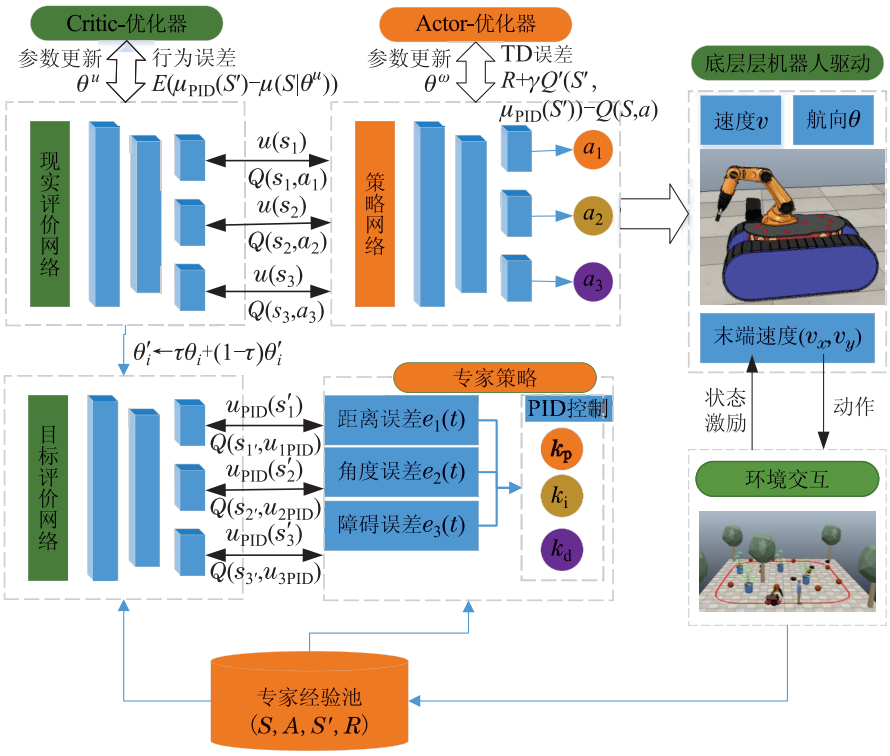


图 3 专家型策略梯度网络结构

Fig. 3 Expert deterministic strategy gradient

Critic当前网络负责评价当前选择动作的优劣, 并且生成当前动作价值

$$Q_{\text{当前}} = Q\left(s_t, a_t|\theta^{\mu}\right). \tag{18}$$

Critic′目标网络负责评价目标动作的优劣, 并且生成目标Q值

$$Q_{\text{目标}} = Q\left(s', \mu^{\text{PID}}\left(s'\right)\right). \tag{19}$$

Critic当前网络以时间差分(TD)的均方差作为更新的损失函数L_{c}, 并且把网络参数每隔一段时间复制给目标网络

$$L_{\text{c}} = \frac{1}{m}\sum_{j=1}^{m}(R + Q(s', \mu^{\text{PID}}(s')) - Q(s_t, a_t|\theta^{\mu}))^2. \tag{20}$$

Actor在专家指导时期, 其目标函数定义为专家指导动作与预测动作的期望误差, 训练网络动作选择策略朝着专家指导的方向更新

$$J = E_{s\sim\rho^{\beta}}\left(\mu^{\text{PID}}\left(s_t\right) - \mu\left(s_t|\theta^{\mu}\right)\right). \tag{21}$$

用于更新Actor当前网络的损失函数为动作误差的均方差

$$L_a = \frac{1}{m}\sum_{j=1}^{m}\left(\mu^{\text{PID}}\left(s_t\right) - \mu\left(s_t|\theta^{\mu}\right)\right)^2. \tag{22}$$

在训练的第2阶段, 专家训练完成后, Actor仍然按照策略梯度方法进行更新. 根据Monte-carlo方法, 在经验池中进行抽样, 对目标函数的梯度进行无偏差估计

$$\nabla_{\theta}\mu J_{\beta}(\mu) \approx \frac{1}{N}\sum i(\nabla_a Q(s, a|\theta^{\varphi})| \cdot \nabla_{\theta^u}\mu(s|\theta^u)), \tag{23}$$

其中: 动作$a = \mu\left(s_t\right)$, 状态$s = s_t$.

当前Actor的网络参数更新后, 并每以软更新的方式将策略参数复制给目标网络, 如式(14) 所示. 此时, 专家指导的PID策略不再工作, 由策略网络继续进行探索和更新, 进一步优化策略性能.

5　试验仿真与验证

本部分试验基于CoppeliaSim机器人物理仿真器, 通过模仿室外工作环境构建训练和测试两种场景, 以验证本文所提出的专家型多智能体算法的性能和机器人跟踪清障能力. 通过对算法的收敛速度、奖励值上升情况、目标跟踪误差和清障成功率等指标的分析, 验证ML-DDPG的性能优势和在陌生环境中的泛化能力.

5.1　试验参数设置

ML-DDPG包括三组Actor-Critic网络. Actor网络和Critic网络采用相同的网络结构. 包括输入层、输出层和两层全连接隐藏层. 激活函数采用线形整流函数ReLU. 输出层采用Sigmoid的函数进行整流, 输出范围为$[-1, 1]$. ML-DDPG 强化学习网络的参数选择如表2所示.

表 2　网络训练参数

Table 2　Network training parameters

参数	介绍	数值
α_a	策略网络学习率	0.001
α_{c}	评价网络学习率	0.001
C	经验回放池容量	5000
Batchsize	单次样本数量	64
γ	折扣因子	0.9995
τ	网络软更新因子	0.01

其中α_a和α_{c}分别表示策略网络和评价网络的模型学习率, 其主要目的是确定策略梯度的更新步长. 以较小的学习率缓慢更新网络权重, 可使损失函数在较小的波动中收敛至稳定值. C表示回放经验池容量, Batchsize表示单次从经验池中抽取的样本数量. 一般的, 经验池容量是单次抽取样本数量的10 倍左右. 实验中设置经验池容量C为5000, Batchsize为64. γ折扣因子代表的是对未来奖励的关注程度. 通常取值在[0.9 1]之间. 在机器人跟踪清障任务中, 选择较大的更加接近于1的值作为折扣因子. τ为当前网络的软更新因子. 为了防止价值函数被高估, 可以选择一个类似于网络学习率的较小值, 去减缓目标网络的更新速度.

5.2　试验场景设置

试验分别设置训练和测试两类环境. 如图4所示.

图4(a)为双环形训练环境, 图4(b)多曲线测试环境. 环境中包含: 清障型机器人、目标人员、各类障碍物以及其它相关对象. 其中, 目标人员沿着固定的轨迹行走以执行相关任务. 清障型机器人需要在跟踪目标人员的同时完成主动障碍清理或者障碍避免任务. 根据场景设置机器人系统的奖励函数为

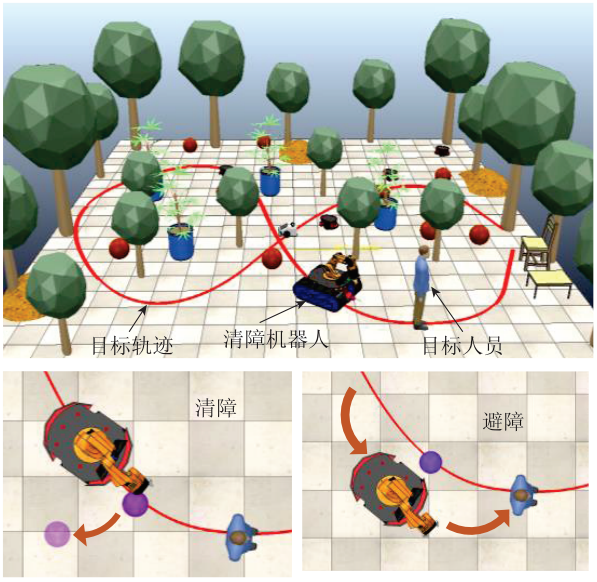
$$\begin{cases} r_1 = -\text{abs}(d - d_0) - \eta \times \text{abs}\left(v_{\text{c}} - v_{\text{o}}\right), \\ r_2 = -\text{abs}(\text{atan}(\dfrac{y - y_t}{x - x_t}) - \theta), \\ r_3 = -\sqrt{(x - x_{\text{o}})^2 + (y - y_{\text{o}})^2 + (z - z_{\text{o}})^2}. \end{cases} \tag{24}$$

其中: r_1为速度Agent的直线跟踪奖励, d_0=1.5 m为最优跟踪距离, v_{c}是机器人速度, v_{o}是目标速度. r_2为转向Agent角度偏移奖励. (x_t, y_t)为跟踪目标坐标, θ 为机器人航向角, r_3为清障Agent的障碍清理奖励, (x, y, z)为机械臂末端的空间坐标, $(x_{\text{o}}, y_{\text{o}}, z_{\text{o}})$为待清理障碍物的空间坐标.

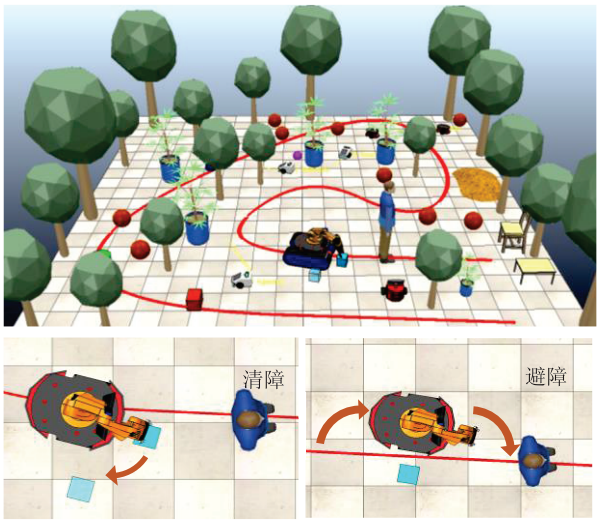
5.3　模型训练

首先, 在仿真试验场景1中对网络模型进行训练. 跟踪机器人以最优的跟踪距离, 保持目标人员位于视角中心, 并对探测到的小型障碍物进行实时清理或避障. 分别使用本文算法ML-DDPG和常规DDPG算法

进行1000回合的训练, 得到两种算法的累计奖励和损失函数收敛情况, 结果如图5和图6 所示.



(a) 场景1: 双环形训练环境



(b) 场景2: 多曲线测试环境

图 4 机器人测试仿真环境

Fig. 4 Training and test simulation environment

图5表示策略网络损失函数的收敛情况, 其中Critic1-3表示ML-DDPG中各个子智能体损失函数随训练轮次的增加的变化趋势, Critic0表示常规DDPG算法的损失函数变化趋势. 通过对比可以发现, 策略网络的损失函数随着训练次数的增加逐渐下降, MLDDPG算法的损失函数在机器人训练280轮过后基本上可以稳定在一个较小值并且达到收敛; 而普通DDPG算法的损失函数Critic0波动较大, 且需要在400轮之后的才逐渐收敛稳定. 图6表示训练过程中每个回合的累计奖励情况. 在多轮训练之后, 两种算方法的均可获得稳定的奖励. 特别的是, 在奖励收敛后, Reward2和Reward0在650回合、800回合和880回合出现了三次明显的下降. 出现这种现象的的主要原因是DDPG和ML-DDPG运动规划策略在动作选择过程中添加了随机噪声N_t所致, 并不会影响策略的整体收敛性. 相比与DDPG方法, 在训练过程中ML-DDPG获得了更高的奖励, 并且奖励的增加速度有明显提高, 从而验证了算法在学习效率和收敛速度上的优越性.

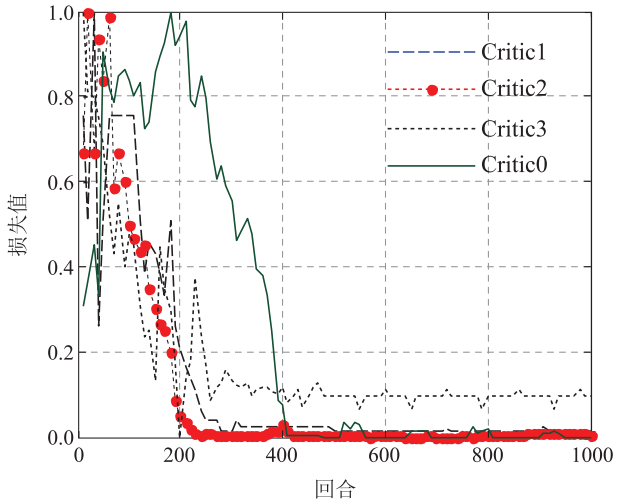


图 5 Critic损失函数对比

Fig. 5 Contrast of critic loss function

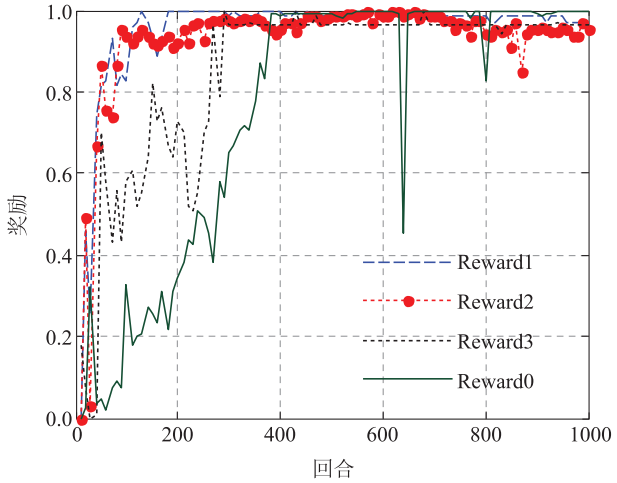


图 6 回合奖励函数对比

Fig. 6 Contrast of reward function

5.4 网络测试

本节将测试基于已收敛的模型的目标跟踪和障碍清理效果. 根据第5.3中完成训练的网络模型, 在场景2控制中移动机器人完成多轮跟踪清障任务, 同时记录测试场景2中机器人的跟踪轨迹, 对比两类算法算法的目标轨迹跟踪效果, 结果如图7所示.

图7代表两种环境中的机器人跟踪路径曲线. 其中, 蓝色虚线代表跟踪目标轨迹, 黑色实线代表在ML-DDPG算法控制下的移动机器人运动轨迹, 红色点划线代表的是DDPG算法的运动轨迹. 通过观察运动轨迹的重合度, 可以判断出ML-DDPG跟踪效果整体优于DDPG算法, 但是在曲率半径较小的转角处,

DDPG算法控制下的跟踪机器人的实际轨迹与目标轨迹出现了偏差, 跟踪效果稍差. 接着, 对清障跟踪机器人的实时距离误差和航向角度误差进行了深入分析, 如图8 所示. 图8表示不同测试环境中的轨迹跟踪误差曲线. 其中, 角度偏差表示机器人航向角与目标方位角的差值; 距离偏差代表的是目标距离与最优距离的差值. 通过两种误差指标的对比可以看出, 无论是是在测试环境还是训练环境之中, ML-DDPG的角度偏差基本稳定在0.1 rad范围内, 距离偏差在0.2 m幅度以内波动. 而DDPG算法无论是角度偏差还是距离偏差的波动值基本上都在本文算法的两倍以上, 结果验证了ML-DDPG算法的对陌生环境模型迁移良好的泛化性和控制的精准性. 同时通过对比运动过程中的机器人的清障成功率可以看出, 由于ML-DDPG在状态输入时考虑到了速度和转向对机械臂清障带来的影响, 加上子智能体的独立控制对机械手动态性能的提高, 在多曲线测试环境中, 机器人的清障成功率分别达到了为98%和97%, 而普通DDPG算法在两种环境中的成功率分别仅有在91%和87%.

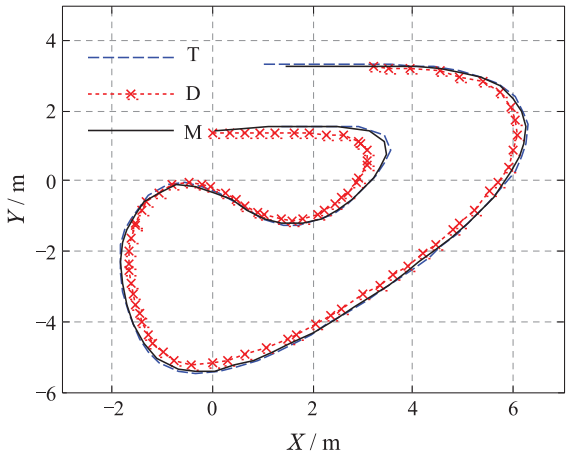
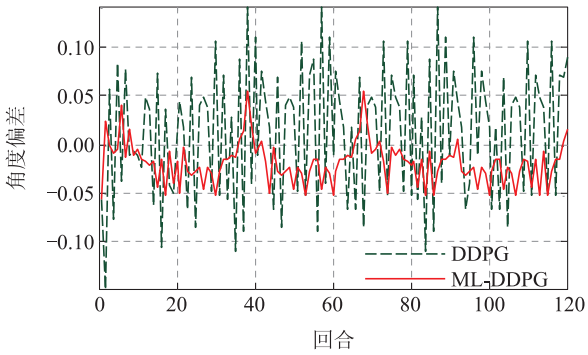


图 7　测试环境中的轨迹跟踪效果

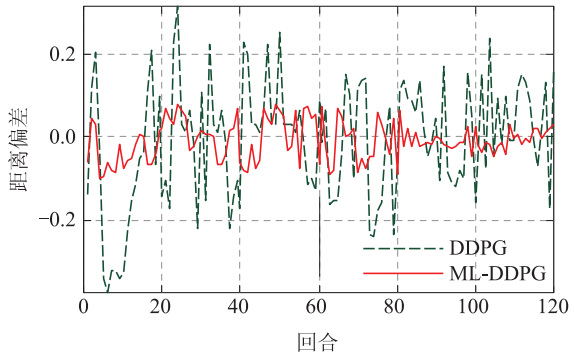
Fig. 7　Trajectory tracking effects in test environments

为了测试多障碍随机环境下机器人的避障、清障以及跟踪效果, 本文构建了一类“无固定路径”的测试环境, 具体场景分布如图9所示.

场景中“目标人员”不再沿着固定的轨迹运动, 而是以一种无约束的方式自由行走至随机出现的“目的地”. 测试试验共设置两种工作模式: 避障模式和清障模式. 在避障模式下, 机器人需要跟随目标在充满障碍的环境中到达随机出现的目的地, 同时通过自身运动避开所有障碍物; 在清障模式下, 机器人需利用机械臂清理开所遇到的障碍物. 在此场景中进行了100轮测试实验, 并统计了遇到的障碍物的总数量、避障(清障)数量和目标跟踪成功率等性能指标, 具体测试结果如表3所示. 从表3中可以看出, 相比于避障模式, 无论是本文ML-DDPG还是常规DDPG算法, 在清障模式下均表现出了较高的跟踪成功率和清障/避障成功率. 机器人可以在测试环境中稳定运行, 并顺利完成跟踪任务; 但是, ML-DDPG的清障(避障)成功率在清障模式下比DDPG提高了5.9%, 在避障模式下提高了2.4%; 其跟踪成功率在两种模式下分别比DDPG方法提升了2% 和5%. 试验结果验证了ML-DDPG方法中多智能体协同控制的优越性, 清障型机器人因此获得了更稳定的运动规划效果.



(a) 多曲线角度跟踪偏差



(b) 多曲线距离跟踪偏差

图 8　训练环境与测试环境

Fig. 8　Tracking errors in different environments

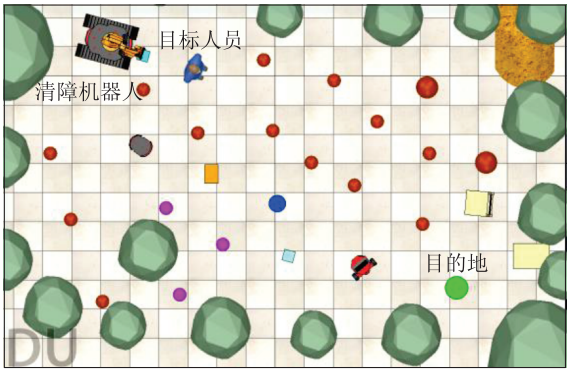


图 9　无固定路径随机测试环境

Fig. 9　Random test environment without trajectory

最后, 本文将深度强化学习运动规划方法与常规的移动机器人避障算法进行比较, 来深入验证清障任务的必要性. 基于图4(b)中所示的的测试环境, 应用RRT运动规划方法控制清障型机器人完成跟踪与避障

任务, 同时记录移动机器人跟踪轨迹, 如图10 中所示. 图中, 红色虚线代表目标人员的行走轨迹, 黑色实线代表机器人运动轨迹. 从图中明显可以看出来机器人为了避开障碍物, 需要不断的调整自己的运动方向, 使自己速度和航向角速度不断发生变化, 同时造成机器人路径总长度不断提高, 特别是面对大负载的多任务机器人, 这不仅会增加目标跟踪的丢失率, 而且会增加本体的能量消耗.

表 3 无路径随机环境中测试结果对比

Table 3 Test results in a random environment

工作模式	方法	障碍数	避/清障数	避/清率/%	目标丢失	跟踪成功率/%
清障	ML-DDPG	580	578	99.6	1	99
清障	DDPG	576	540	93.7	3	97
避障	ML-DDPG	600	560	93.3	10	90
避障	DDPG	605	550	90.9	15	85

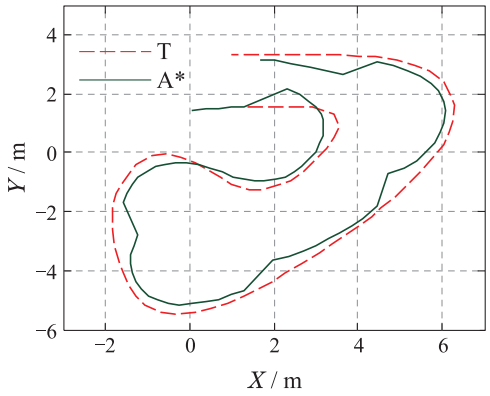


图 10 跟踪机器人避障轨迹

Fig. 10 The obstacle avoidance trajectory of mobile robot

在测试仿真环境中, 5自由度机械臂的工作额定功率为P_1, 静止状态下的功率消耗为P_2, 机器人底盘移动消耗的功率为P_3. 清障机器人完成一个回合试验测试所消耗的总的能量为

$$E_{总} = P_1 * t_1 + P_2 * t_2 + P_3 * t_3, \qquad (25)$$

其中: t_1为机械臂工作时间, t_2为机械臂姿态保持时间, t_3为机器人底盘移动工作时间.

接着, 笔者分别记录了ML-DDPG, DDPG以及RRT3种方法在每个回合中的运动总路程$l_{总}$以及机械臂的工作时间t_1, 并绘制变化曲线图, 如图11所示.

根据实验设置, 机器人地盘移动的平均消耗功率为300 W, 机械臂有清障动作时的平均消耗功率为100 W, 无清障动作时的消耗功率为40 W, 机器人地盘的平均运行速度为1 m/s. 3种控制模式下机械臂及地盘的平均工作时间如表4所示.

根据能量消耗计算式(25)和表4中的平均工作时间, 分别计算出机器人3种控制模式下的能量消耗平均值, 所得详细结果如表5所示.

从图11中可以看出, 在清障模式下工作的MLDDPG运动规划方法具有最小的运动总路程和较低的机械臂工作时间. 由于普通RRT运动规划方法不参与清障工作, 机械臂工作时间为0 s, 但是机器人产生了最大的工作总路程. 然后, 通过表5中数据对比可以看出, 在清障模式下, 无论是ML-DDPG还是DDPG方法, 机器人总能量消耗均小于RRT避障方法, 这是因为较大的工作总路程加剧了RRT运动规划方法机器人本体的能量消耗, 并且这种消耗会随着障碍物的增多和机器人本体负载的增加而不断提高. 进一步的, 对比ML-DDPG和DDPG的之间测试结果可以看出, 由于ML-DDPG方法在机器人目标跟踪过程中的动态调误差较小, 机械臂可以较准确的清理所遇障碍物, 其机器人本体能量消耗和机械臂能量消耗均小于常规DDPG方法, 总能量消耗处于最低值.

表 4 不同算法机器人底盘及机械臂工作时间(单位: s)

Table 4 Working time for robot body and arm of different algorithms (unit: s)

方法	Arm-清障	Arm-无动作	机器人
ML-DDPG	6.1	14.5	20.6
DDPG	8.2	15.6	23.8
RRT避障	0	0	29.5

表 5 不同算法平均能量消耗值(单位: J)

Table 5 Energy consumption of different algorithms(unit: J)

方法	Arm-清障	Arm-无动作	机器人	总能量
ML-DDPG	610	580	6180	7370
DDPG	820	624	7140	8584
RRT避障	0	0	8850	8850

综合以上的试验结果, 通过分析与对比不同运动规划方法之间的奖励值、收敛速度、目标跟踪误差、障碍清理能力和能量消耗状态的差异, 可以明显得出, ML-DDPG运动规划方法在多障碍未知环境中的目标跟踪与障碍清理任务上有较大的性能优势. ML-DDPG不仅提高了机器人跟踪成功率, 而且降低了机器人

整体系统能量消耗. 在运动策略的优化方面, ML-DDPG因为采用了专家策略引导的多智能体结构, 模型训练的收敛速度得到了提高, 机器人运动控制的跟踪误差有明显的降低. 多轮综合性试验验证了ML-DDPG在不同场景之间的具有良好的模型迁移能力, 适用于多任务型的机器人控制任务.

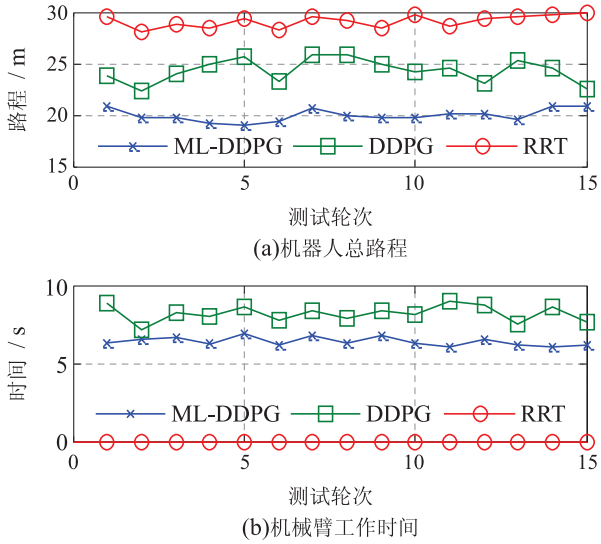


图 11　总路程及机械臂工作时间

Fig. 11　Total distance and working time of the manipulator

5.5　真实试验环境中的可行性分析

基于Multi-agent专家型策略梯度的目标跟踪策略在仿真环境中展现了较好的效果. 但是, 将此策略应用于真实环境中, 可能还面临着以下挑战:

1) 策略模型从仿真环境向真实环境迁移的困难.

由于真实环境中存在设备安全性和机械可靠性的限制, 清障型跟踪机器人需要首先在仿真环境中训练出策略模型, 然后再将模型迁移到物理样机上进行测试. 但是, 仿真和真实试验条件存在较大的差距, 迁移后的模型可能会出现特征匹配不准确, 动作输出次优等问题, 其策略参数需要进行大量人工调整工作才能正常运行. 为了解决这一问题, 未来需要进一步提升仿真环境的真实度, 提前将任务环境中的约束差异添加到仿真环境中, 同时在仿真环境中添加模拟真实环境的摩擦、震动、噪声等干扰项, 使训练出的运动策略更好的匹配真实环境同时具有更好的泛化性, 让策略模型可以顺利的从仿真环境中迁移至真实环境.

2) 传感器数据信息不完整, 通讯数据流延迟.

在真实试验场景中, 由于光照变化、障碍遮挡、电磁干扰和温度波动等因素的影响, 容易出现传感器采集数据不全, 信息传输延迟等问题. 不稳定的数据输入将给清障型跟踪机器人的决策系统带来巨大困扰, 从而引起跟踪目标识别率降低、动作决策延迟和清障避障失败等现象. 面对此类问题, 可以采用多传感器融合的数据感知方法, 去大幅度提高传感器数据的可靠性. 多重同型的传感器可以保证数据传输的冗余性, 多类异型的传感器通过信息融合, 可增强数据采集模块对环境变化的鲁棒性, 保证输入信息的完整度.

3) 移动端处理器数据运算能力不足.

在仿真环境中, 通常采用PC端高性能处理器和显卡作为数据处理单元和运算中心, 设备可以轻松的处理高维数据输入和大规模的网络模型. 但是, 在真实环境中, 由于移动设备自身硬件条件的限制, 移动端难以完成高速的数据运算工作, 甚至容易出现设备过热、死机、模型卡顿等问题. 对此, 未来可以从两个方面来入手解决此类问题. 第一, 增强踪机器人设备终端性能, 采用更加先进专业的机器学习开发套件, 如, 英伟达Jetson系列控制板. 第二, 优化多智能体强化学习网络架构, 设计高效的机器人运动规划策略模型, 寻求数据处理准确性和模型流畅性的最佳匹配点, 在保证跟踪清障机器人完成目标任务的同时, 最大限度的轻量化强化学习策略模型.

6　结论

本文面向清障型机器人的目标跟踪任务, 提出了一种基于多智能体专家型策略梯度的运动规划方法. 该方法以深度确定性策略为基础, 建立了以单元任务目标为导向的分布式多智能体单元框架, 摒弃了传统强化学习中依赖单智能体去完成多输入多输出的工作模式, 降低了单个智能体网络结构的复杂度, 以并行的训练方式提高了模型的训练效率; 针对强化学习前期无效探索过多、训练效率低的问题, 增加了以专家型知识引导的动作选择机制, 提升了模型目标动作选择的可靠性, 同时降低了目标动作价值被过分高估的问题; 为了增强深度强化学习的目标感知能力, 利用多层卷积神经网络模块对强化学习的图像状态输入进行预先处理, 以特定的状态向量输入到强化学习网络, 降低了策略网络对状态信息的解析压力. 最后, 分别在的训练环境和测试环境中对ML- DDPG的可行性进行了验证, 并与传统的DDPG算法进行了性能指标对比. 试验结果表明, ML-DDPG较于常规DDPG算法的收敛速度有了接近30%的提升, 目标跟踪误差降低了50%, 清障成功率提升5.9%, 同时在测试环境中证明了策略模型良好的迁移和泛化能力. 未来, 我们将尝试把ML-DDPG算法迁移至真实机器人上, 在真实环境中更好的完成机器人目标跟踪和自主导航等协同作业任务.

参考文献:

[1] LUO Xin, DING Xiaojun. Research and prospective on motion planning and control of ground mobile manipulators. *Journal of Harbin Institute of Technology*, 2021, 53(1): 1 – 15.

(罗欣, 丁晓军. 地面移动作业机器人运动规划与控制研究综述. 哈尔滨工业大学学报, 2021, 53(1): 1 – 15.)

[2] WANG Changshun, WANG Dan, PENG Zhouhua. Coordinated formation control of car-like mobile robots guided by parameterized single path. *Control Theory & Applications*, 2021, 38(7): 1124 – 1132.
(王常顺, 王丹, 彭周华. 单路径导引的车式移动机器人协同编队控制. 控制理论与应用, 2021, 38(7): 1124 – 1132.)

[3] SUN Huihui, HU Chunhe, ZHANG Junguo. Deep reinforcement learning for motion planning of mobile robots. *Control and Decision*, 2021, 36(6): 1281 – 1292.
(孙辉辉, 胡春鹤, 张军国. 移动机器人运动规划中的深度强化学习方法. 控制与决策, 2021, 36(6): 1281 – 1292.)

[4] CHENG C X, SHA Q X, HE B, et al. Path planning and obstacle avoidance for AUV: A review. *Ocean Engineering*, 2021, 235: 109355.

[5] YANG W L, WU P, ZHOU X Q, et al. Improved artificial potential field and dynamic window method for amphibious robot fish path planning. *Applied Sciences-Basel*, 2021, 11(5): 1 – 6.

[6] WEN N F, ZHANG R B, LIU G Q, et al. Online planning low-cost paths for unmanned surface vehicles based on the artificial vector field and environmental heuristics. *International Journal of Advanced Robotic Systems*. 2020, 17(6): 172988142096907.

[7] YAO L J, PITLA S K, ZHAO C, et al. An improved fuzzy logic control method for path tracking of an autonomous vehicle. *Transactions of the ASABE*, 2020, 63(6): 1895 – 1904.

[8] WANG M C. Real-time path optimization of mobile robots based on improved genetic algorithm. *Proceedings of the Institution of Mechanical Engineers Part I–Journal of Systems and Control Engineering*, 2021, 235(5): 646 – 651.

[9] LIU X H, ZHANG D G, ZHANG J, et al. A path planning method based on the particle swarm optimization trained fuzzy neural network algorithm. *Cluster Computing–the Journal of Networks Software Tools and Applications*, 2021, 24(3): 1901 – 1915.

[10] ZHU Zhibin, WANG Fuyong, YIN Yanhui, et al. Consensus of discret-time multi-agent system based on Q-learning. *Control Theory & Applications*, 2021, 38(7): 997 – 1005.
(朱志斌, 王付永, 尹艳辉, 等. 基于Q-learning的离散时间多智能体系统一致性. 控制理论与应用. 2021, 38(7): 997 – 1005.)

[11] YU Lingi, SHAO Xuanya, LONG Ziwei, et al. Inelligent land vehicle model transfer trajectory planning method of deep reinforcement learning. *Control Theory & Applications*, 2019, 36(9): 1409 – 1422.
(余伶俐, 邵玄雅, 龙子威, 等. 智能车辆深度强化学习的模型迁移轨迹规划方法. 控制理论与应用, 2019, 36(9): 1409 – 1422.)

[12] VARGHESE N V, MAHMOUD Q H. A survey of multi-task deep reinforcement learning. *Electronics*, 2020, 9(9): 1363 – 1370.

[13] WANG H N, LIU N, ZHANG Y Y, et al. Deep reinforcement learning: A survey. *Frontiers of Information Technology Electronic Engineering*,2020, 21(12): 1726 – 1744.

[14] HUANG X Q, DENG H, ZHANG W, et al. Towards multi-modal perception-based navigation: A deep reinforcement learning method. *IEEE Robotics and Automation Letters*, 2021, 6(3): 4986 – 4993.

[15] KULHANEK J, DERNER E, BABUSKA R. Visual navigation in real-world indoor environments using end-to-end deep reinforcement learning. *IEEE Robotics and Automation Letters*, 2021, 6(3): 4345 – 4352.

[16] MORAD S D, MECCA R, POUDEL R, et al. Embodied visual navigation with automatic curriculum learning in real environments. *IEEE Robotics and Automation Letters*, 2021, 6(2): 683 – 690.

[17] SAMSANI S S, MUHAMMAD M S. Socially compliant robot navigation in crowded environment by human behavior resemblance using deep reinforcement learning. *IEEE Robotics and Automation Letters*, 2021, 6(3): 5223 – 5230.

[18] DEJESUS J C, KICH V A, KOLLING A H, et al. Soft actor-critic for navigation of mobile robots. *Journal of Intelligent Robotic Systems*, 2021, 102(2): 1 – 11.

[19] SHI H B, SHI L, XU M, et al. End-to-end navigation strategy with deep reinforcement learning for mobile robots. *IEEE Transactions on Industrial Informatics*, 2020, 16(4): 2393 – 2402.

[20] YAN N, HUANG S B, KONG C. Reinforcement learning-based autonomous navigation and obstacle avoidance for USVs under partially observable conditions. *Mathematical Problems in Engineering*, 2021, DOI: 10.1155/2021/5519033.

[21] MALDONADO-RAMIREZ A, RIOS-CABRERA R, LOPEZ-JUAREZ I. A visual path-following learning approach for industrial robots using DRL. *Robotics and Computer-Integrated Manufacturing*, 2021, 71(6419): 102130.

[22] WANG D, DENG H B. Multirobot coordination with deep reinforcement learning in complex environments. *Expert Systems with Applications*, 2021, 180: 115128.

[23] BI Weiping, ZHANG Huan,QU Zhenlin, et al. Design of autonomous following system for master-slave vehicles operating in orchard based on binocular stereo vision. *Journal of Hunan Agricultural University (Natural Sciences)*, 2016, 42(3): 344 – 348.
(毕伟平, 张欢, 瞿振林, 等. 基于双目视觉的主从式果园作业车辆自主跟随系统设计. 湖南农业大学学报: 自然科学版, 2016, 42(3): 344 – 348.)

[24] DEND Xiaohao, HOU Jin,TAN Guanghong, et al. Multi-objective vehicle following decision algorithm based on reinforcement learning. *Control and Decision*, 2021, 36(10): 2497 – 2503.
(邓小豪, 侯进, 谭光鸿, 等. 基于强化学习的多目标车辆跟随决策算法. 控制与决策, 2021, 36(10): 2497 – 2503.)

[25] GUO F, CHENG G, PANG Y. Explicit dynamic modeling with joint friction and coupling analysis of a 5-DOF hybrid polishing robot. *Mechanism and Machine Theory*, 2022, 167: 104509.

[26] BAEK J, JUN H, PARK J, et al. Sparse variational deterministic policy gradient for continuous real-time control. *IEEE Transactions on Industrial Electronics*, 2021, 68(10): 9800 – 9810.

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

[28] MAGALHAES S A, CASTRO L, MOREIRA G, et al. Evaluating the single-shot MultiBox detector and YOLO deep learning models for the detection of tomatoes in a greenhouse. *Sensors*, 2021, 21(10): 3569.

作者简介:

孙辉辉 博士研究生, 目前研究方向为多智能规划与决策、深度强化学习、机器人智能控制等, E-mail: cumtsunhui@126.com;

胡春鹤 副教授, 硕士生导师, 目前研究方向为无人机智能控制、多智能体协作强化学习等, E-mail: huchunhe@bjfu.edu.cn;

张军国 教授, 博士生导师, 目前研究方向为深度学习、多目标识别、机器人智能控制等, E-mail: zhangjunguo@bjfu.edu.cn.