智能系统学报
CAAI TRANSACTIONS ON INTELLIGENT SYSTEMS

融合专家纠偏策略的移动机器人动态环境避障方法

田顺钰, 欧阳勇平, 魏长赟

引用本文:
田顺钰, 欧阳勇平, 魏长赟. 融合专家纠偏策略的移动机器人动态环境避障方法[J]. 智能系统学报, 2024, 19(6): 1492-1502.
TIAN Shunyu, OUYANG Yongping, WEI Changyun. Collision avoidance approach with heuristic correction policy for mobile robot navigation in dynamic environments[J]. *CAAI Transactions on Intelligent Systems*, 2024, 19(6): 1492-1502.

在线阅读 View online: https://dx.doi.org/10.11992/tis.202304056

DOI: 10.11992/tis.202304056
网络出版地址: https://link.cnki.net/urlid/23.1538.tp.20240909.1120.009

融合专家纠偏策略的移动机器人动态环境避障方法

田顺钰, 欧阳勇平, 魏长赟
(河海大学 机电工程学院, 江苏 常州 213251)

摘　要: 基于深度强化学习(deep reinforcement learning, DRL)的移动机器人无地图导航技术备受机器人和相关研究领域的关注, 在非结构化环境中避免与动态障碍物的碰撞是需要解决的重要难题。为此提出一种融合专家纠偏策略的机器人自主导航 DRL 方法, 该算法将 24 线激光雷达传感器信息、目标位置信息和机器人速度信息作为深度强化学习的输入, 并输出控制机器人的动作指令。实验结果表明, 相较于其他算法, 该算法可以在保证安全的前提下以更短的距离和时间到达目标。同时将所提出的方法部署在真实机器人上, 验证和评估算法的性能, 为机器人动态环境避障导航提供一种技术参考。

关键词: 移动机器人; 深度强化学习; 机器人导航; 非结构环境; 动态避障; 专家纠偏策略; 自学习; 端到端
中图分类号: TP273+.2　　**文献标志码**: A　　**文章编号**: 1673−4785(2024)06−1492−11

中文引用格式: 田顺钰, 欧阳勇平, 魏长赟. 融合专家纠偏策略的移动机器人动态环境避障方法 [J]. 智能系统学报, 2024, 19(6): 1492–1502.
英文引用格式: TIAN Shunyu, OUYANG Yongping, WEI Changyun. Collision avoidance approach with heuristic correction policy for mobile robot navigation in dynamic environments[J]. CAAI transactions on intelligent systems, 2024, 19(6): 1492–1502.

Collision avoidance approach with heuristic correction policy for mobile robot navigation in dynamic environments

TIAN Shunyu, OUYANG Yongping, WEI Changyun
(College of Mechanical and Electrical Engineering, Hohai University, Changzhou 213251, China)

Abstract: Mapless navigation for mobile robots based on deep reinforcement learning (DRL) has received increasing attention from robotics and related research fields. The major challenge in mapless navigation is collision avoidance of dynamic obstacles in unstructured environments. Therefore, this paper proposes a DRL algorithm that incorporates a heuristic correction policy for robot autonomous navigation. The algorithm utilizes information from a 24-line laser radar sensor, target location, and robot velocity as inputs for DRL to generate action commands that regulate the robot's motion. Experimental results demonstrate that, compared to other algorithms, the proposed approach can reach the target more efficiently in terms of distance and time while ensuring safety. Moreover, the algorithm is implemented in a real robot to verify and evaluate its performance, providing a technical reference for collision avoidance during its navigation in dynamic environments.

Keywords: mobile robots; deep reinforcement learning; robot navigation; non-structural environment; dynamic collision avoidance; heuristic correction policy; self-learning; end-to-end

近年来, 自主导航机器人的应用范围和场景不断扩大[1-4]。机器人的自主导航需要找到最佳的导航路径, 并在有人围绕或接近机器人时安全避开障碍物。传统的避障导航算法可根据环境中先验地图信息是否已知分为基于地图的避障导航[5-6]和无地图避障导航[7-8]。在已知地图信息下, 机器人避障导航的经典算法有 A*算法[9]、D*算法[10]和 RRT*算法[11]等, 需要使用实时定位和建图(simultaneous localization and mapping, SLAM)[12-15]

收稿日期: 2023−04−27.　**网络出版日期**: 2024−09−09.
基金项目: 国家自然科学基金项目(52371275).
通信作者: 魏长赟. E-mail: c.wei@hhu.edu.cn.

等技术预先构建地图，再进行路径规划和避障导航。然而，基于地图的导航方法构建地图成本昂贵，并且复杂环境下需要多传感器数据融合，增加了导航过程的复杂性。在许多应用场景中，机器人无法创建精确的全局环境地图，只能根据目标的相对位置信息和机载传感器对局部环境感知来控制运动。近年来，深度强化学习（deep reinforcement learning, DRL）算法在无地图机器人导航任务上取得了显著的成就，相关学者提出了从状态空间到动作空间的策略映射[16-20]，Xue 等[16]提出了一种结合柔性 Actor-Critic 网络和自动课程学习的方法，构建基于激光雷达和 RGB 相机的自主导航策略映射。然而，该算法未考虑动态环境下的避障策略，且未应用于实体机器人进行鲁棒性测试。Han 等[17]提出了一种自状态注意单元算法，使用激光雷达和单目摄像头来实现机器人的避碰策略，但其算法的收敛速度慢，训练周期长，需要大量的时间进行算法训练。因此，设计一种能够使机器人在动态环境中安全、高效地从起始位置导航到目标位置的自主导航避障策略具有重要意义。

本研究提出了一种新的采用专家纠偏策略（heuristic correction policy, HCP）的深度强化学习算法，并引入优先经验回放机制，不仅提高了算法的训练速度，还提升了机器人避障导航的成功率与轨迹效率。同时本研究基于 ROS 平台搭建了机器人控制模型，在 Gazebo 仿真环境下进行算法训练与测试，并迁移至实体机器人，在现实环境下测试避障导航方法的可用性，实现了移动机器人的避障导航任务。

1　深度强化学习模型

1.1　无地图导航避障模型

多机器人的导航避障问题是指在动态环境中，机器人通过传感器收集周围环境信息，通过控制模型的处理，执行导航避障算法输出的动作指令，有效地规避障碍物并前往目标区域执行任务。其中，单机器人不仅需要躲避静态障碍物，还需要避免与其他机器人或动态障碍物碰撞。因此，在未知环境先验信息的前提下实现多机器人的导航避障功能仍是一大难题。

为实现机器人导航避障，其必须具有对外界环境的感知能力，通过车载传感器进行获取周围未知环境的信息，感知机器人周围环境中存在的障碍物的大小、形状和位置等信息。为合理表述机器人周围的环境信息，本研究构建了不依赖于地图的机器人导航避障模型，机器人仅安装二维雷达传感器，以获取环境信息。具体模型如图 1 所示。连接到机器人上的红色线条表示了激光雷达距离测量光束，根据绿色箭头方向顺序获取射线测量值；五角星代表目标位置；蓝色箭头表示当前时刻前进方向；橙色箭头表示机器人下一时刻速度指令；矩形方块和大箭头分别表示静态、动态障碍物。

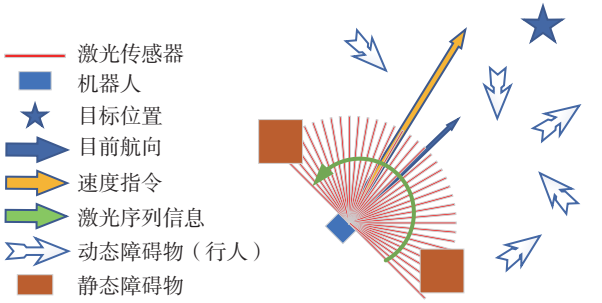

图 1　机器人导航避障模型

Fig. 1　Robot navigation and collision avoidance model

根据上述避障模型的构建，定义几个已知参数。定义：t时刻机器人与目标点之间的相对位置为 g_t，t 时刻机器人观测到的激光雷达测量值 $l_t=(l_1,l_2,\cdots,l_n)$，n 为雷达射线数。假设机器人在环境中获取的信息为部分观测状态，即超出雷达射线范围无法观测。

1.2　马尔可夫决策过程模型

马尔可夫决策过程 (Markov decision process, MDP) 作为强化学习的重要基础，在机器人控制领域得到广泛应用[21-22]。MDP 可由五元素的数组形式表示为(S,A,π,R,G)，S 是状态空间，表示智能体在环境中所有的状态集合，即 $s_t\in S$；A是动作空间，表示智能体在环境中所有动作的集合，即 $a_t\in A$；π是策略，表示智能体状态的条件转移概率；R是奖励函数，表示t时刻下的智能体在状态为 $s_t=s$时，采取动作$a_t=a$所对应的奖励值为R_{t+1}；G是收获函数，表示智能体在环境中所处状态为 $s_t=s$时，经过策略π，采取动作$a_t=a$的价值。

根据 MDP 模型的定义，将机器人无地图导航避障问题转换为部分可观测的马尔可夫决策过程模型（partially-observable Markov decision process, POMDP）。用 MDP 模型的数组可以表示为(O,A,π,R,G)，O代表机器人在环境中获取的部分观测状态空间。根据上述马尔可夫决策模型的定义，基于无地图的导航避障模型可以描述为寻找一个用于求解最优动作的策略函数：

$$a_t\sim\pi_\theta(o_t,a_{t-1}) \tag{1}$$

式中：o_t为t时刻机器人的观测状态，包括雷达射线测量值l_t以及与目标点的相对距离g_t；a_{t-1}指$t-1$时刻机器人的动作，包括机器人的线速度与角速度。机器人将观测状态和上一时刻动作输入

策略函数中，通过策略函数采样输出的最优动作指令移动。

马尔可夫决策过程可以实现机器人与环境之间的交互，从而根据奖励函数的设置来训练机器人学习避障导航策略。基于马尔可夫决策过程，可以构建深度强化学习网络，以解决机器人无地图导航避障问题。

2　融合专家纠偏策略的孪生延迟深度确定性策略梯度改进算法

在现有的融合专家纠偏策略的孪生延迟深度确定性策略梯度（twin delayed deep deterministic policy gradient, TD3）算法[23]基础上，针对优先采样机制以及采样模型进行优化，并增加专家纠偏策略（heuristic correction policy, HCP），构建了专家纠偏−孪生延迟深度确定性策略梯度算法（HCP-TD3）。

2.1　TD3 算法框架

TD3 算法是一种基于 Actor-Critic 架构[24]的深度强化学习（DRL）算法，适用于具有连续观察和动作空间的智能体。TD3 算法结合了深度确定性策略梯度（deep deterministic policy gradient, DDPG）算法和双 Q 学习，在许多连续控制任务中取得了不错的表现，其具体算法框架如图 2 所示。

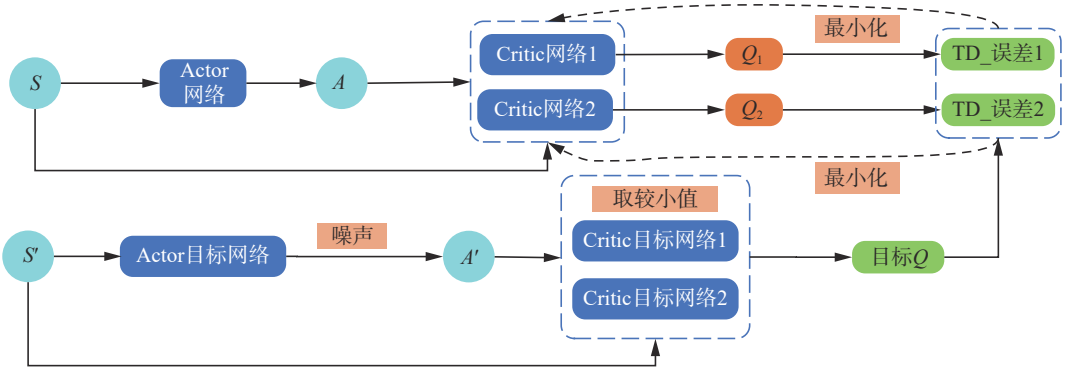

图 2　孪生延迟深度确定性策略梯度算法

Fig. 2　TD3 algorithm

TD3 整体框架中，采用双 Q 网络的方式对 Critic 网络 (C 网络) 优化，以解决 C 网络中 Q 高估的问题。与 DDPG 算法的神经网络参数更新方式不同，TD3 算法采用 Actor 网络 (A 网络) 延迟更新方式，提高机器人的训练稳定性，同时在目标网络中，加入噪声以增加算法的稳定性和鲁棒性。

针对机器人无地图避障导航问题，需要根据马尔可夫决策过程为 TD3 算法设计 4 个重要部分：状态空间、动作空间、奖惩函数以及 A 网络和 C 网络框架。

状态空间包括机器人雷达传感器观测信息 l_t、机器人自身速度 $\boldsymbol{v}_t$ 以及机器人与目标相对距离 $\boldsymbol{g}_t$。其中，雷达传感器信息 l_t 是 24 维雷达数据经过滤波和归一化处理得到的，速度 $\boldsymbol{v}_t$ 是二维向量，包括线速度和角速度，相对目标距离 $\boldsymbol{g}_t$ 是二维向量，由相对目前位置的距离值和相对角度表示。

动作空间包括 2 个速度，机器人移动线速度 l_v 和旋转角速度 a_v，其中，线速度范围为 $l_v \in (0,1)$，角速度范围为 $a_v \in (-\pi/2, \pi/2)$。

奖惩函数由 4 部分组成，目的是鼓励机器人朝着目标方向快速、准确运动，其表示式为

$$r = r_{\mathrm{d}} + r_{\mathrm{c}} + r_{\mathrm{va}} + r_{\mathrm{vl}} \tag{2}$$

式中：r 为总奖励值，r_{d} 为距离奖励值，r_{c} 为碰撞奖励值，r_{va} 为角速度奖励值，r_{vl} 为线速度奖励值。

r_{d} 计算公式为

$$r_{\mathrm{d}} = \begin{cases} r_{\mathrm{ar}}, & d_g \leqslant d_{g\min} \\ \Delta d_g, & \text{其他} \end{cases} \tag{3}$$

式中：d_g 为机器人与目标的相对位置距离值；$d_{g\min}$ 为最小阈值。如果 d_g 小于阈值 $d_{g\min}$，表示机器人已经抵达目标区域，同时获得一个奖励值 r_{ar}；Δd_g 为当前时刻机器人与目标之间的相对距离 d_t 和上一时刻机器人与目标之间的相对距离 d_{t-1} 的差值。

此外，碰撞奖励值 r_{c} 计算公式为

$$r_{\mathrm{c}} = \begin{cases} -\exp(-k_l(l_{\min} - o_l)/l_{\max}), & \bar{l}_{\min} < k_{\mathrm{d}} \\ 0, & \text{其他} \end{cases} \tag{4}$$

式中：$l_{\min} = \min(l_1, l_2, \cdots, l_n)$ 为雷达传感器中雷达射线最短距离值；k_l 和 o_l 为常量，用于确定曲线形状的增益和距离偏移；$\bar{l}_{\min}$ 为 $l_{\min}$ 归一化的值；k_{d} 为一个碰撞阈值，如果 $\bar{l}_{\min}$ 小于 k_{d}，产生负碰撞奖励值。

角速度奖励值 r_{va} 计算公式为

$$r_{va}=\begin{cases} r_a, & |a_v|>k_{av}|a_{v\max}| \\ 0, & \text{其他} \end{cases} \tag{5}$$

式中：r_a为角速度惩罚值；$a_{v\max}$为最大角速度值阈值；k_{av}为一个系数，用于调节角速度的最大阈值范围。如果角速度$|a_v|$大于$k_{av}|a_{v\max}|$，则给其惩罚值。

线速度奖励值计算公式为

$$r_{vl}=\begin{cases} r_l, & l_v<l_{v\min} \\ 0, & \text{其他} \end{cases} \tag{6}$$

式中：r_l为线速度惩罚值，$l_{v\min}$为最小线速度值阈值。如果线速度l_v小于$l_{v\min}$，则给其惩罚值。设置 2 个速度的惩罚值是为了能够让机器人在训练时，避免旋转过快和停滞不前。

TD3 框架下 A 网络与 C 网络具体的网络框架构建，如图 3 和图 4 所示。

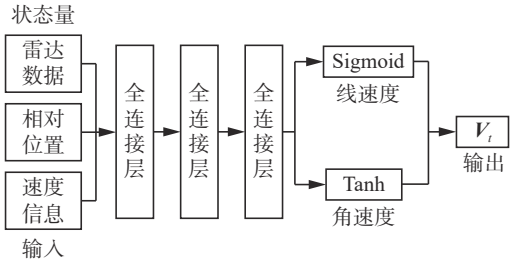

图 3　A 神经网络框架

Fig. 3　Actor neural network

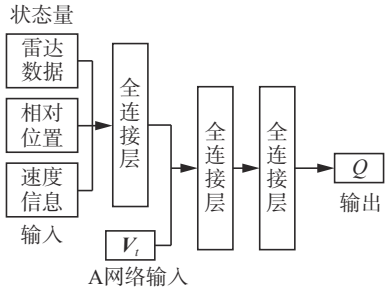

图 4　C 神经网络框架

Fig. 4　Critic neural network

A 网络的输入是多维数据向量，包括 24 维激光数据、二维相对目标位置信息和二维机器人速度信息。输入数据由 3 个完全相同的全连接层连接，每个全连接层都具有 500 个神经网络节点，多维状态信息经过 3 个全连接层之后，再分别通过 Sigmoid 函数和 Tanh 函数计算产生线速度和角速度，最后合并形成速度命令。合并之后的速度命令将作为 C 网络的动作输入，但与 A 网络的状态输入不同的是，C 网络的状态输入需要连接到第 2 层全连接层。最后，C 网络通过线性激活函数生成目标 Q。

2.2　改进优先经验回放策略

经验采样机制是 TD3 算法提高训练速度的手段，通常情况下，经验是从内存缓冲区（记忆池）中均匀采样，但不考虑其意义，因此采样效率较低。为了避免对历史数据进行统一采样，采用优先经验回放 (priority experience replay, PER) 算法[25]，以保证对重要历史数据进行更频繁的采样，从而加速算法训练速度。PER 机制作为 TD3 算法的重要组成部分，解决了移动机器人在各种场景应用中遇到的样本多样性损失问题。

经验回放机制过程分为 2 部分：经验的存放和经验的抽取方式。针对具有经验池的强化学习算法来说，如果选取不到好的经验，算法难以学到有用信息，严重降低算法的整体学习效率。对此，需要对每条经验进行重要性排序，优先选取价值最高的经验进行采样，即优先经验回放策略。

经典 PER 模型的衡量标准是 TD 误差，该误差损失函数部分可以表示为

$$L_{\text{oss}}=\mathrm{E}[Q_{\text{target}}(s,a)-Q(s,a)]^2 \tag{7}$$

式中：L_{oss}为误差损失函数，$Q_{\text{target}}(s,a)$为从目标 C 网络输出的 Q，$Q(s,a)$为从当前 C 网络输出的 Q。

但是在奖励稀疏的情况下，使用基于 TD 误差的方法对其进行样本采样时，会出现算法性能下降以及优先级存在的多样性损失问题。为解决此问题，本研究在原本的采样机制上使用了重要性采样方法和随机优先级方法。

重要性采样是统计学中用于估计各种分布性质所采用的方法，其本质是用一种分布逼近另一种分布。根据重要性采样定义，假设要求得在目标x下函数$f(x)$分布，$f(x)$是一个很复杂的概率密度函数，无法对目标x直接进行采样，因为直接采样方式会使求解过程难度剧增。对此，本研究采用蒙特卡罗方法对其进行求解。因为样本分布存在多样性，在求取期望值的时候，假如取样过少就会存在差异很大的结果，出现估计值$f(x)q(x)$与期望值$f(x)p(x)$不一致的问题，如图 5 所示。

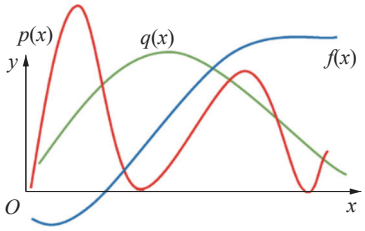

图 5　重要性采样示意

Fig. 5　Importance sampling diagram

图 5 中：$q(x)$表示逼近分布，$p(x)$表示目标样本分布，$f(x)$表示的是所求分布。在样本分布非常复杂时，由重要性采样方法定义，使用一个分布去逼近这个复杂的分布。根据重要性采样，求

解分布$f(x)$，得到重要性采样期望值公式：

$$\mathrm{E}_{x\sim p(x)}[f(x)]=\int_x f(x)p(x)\mathrm{d}x=\int_x f(x)\frac{p(x)}{q(x)}q(x)\mathrm{d}x=\mathrm{E}_{x\sim q(x)}\left[f(x)\frac{p(x)}{q(x)}\right]\simeq\frac{1}{n}\sum_{i=1}^{n}f(x)\frac{p(x)}{q(x)} \tag{8}$$

式(8)表示根据新的近似分布$q(x)$进行采样，最后直接计算所求目标样本在 2 个分布上的值，即可得到所求分布$f(x)$的近似期望值。

为了将 PER 模型与强化学习算法进行配合，需要设计重要性采样权重，可表示为

$$w_i=\left(\frac{1}{B}\cdot\frac{1}{P(i)}\right)^k \tag{9}$$

式中：w_i为重要性采样权重，其作用是表示每条经验的重要性；B为强化学习算法中训练经验的批次数量；k为设计的一个超参数，$k\in[0,1]$，表示的是采样性权值，其作用是抵消经典 PER 算法收敛产生的影响。

为了增加经验中的有用信息，避免采样单一性，使用随机优先级方法。考虑到 TD 误差是由 C 网络中存在的损失函数进行计算得出，因此通过 A 网络的损失函数对经验优先级进行优化处理，得到新的计算经验优先级公式：

$$\begin{aligned}p_i&=\delta_i^n+\lambda\left|\nabla_{\mathrm{A}}Q(s_t,a_t|\theta^{Q_a})\right|^n+\varepsilon\\\delta_i&=R_i+\max_{a'}\gamma Q_{\mathrm{target}}(s',a')-Q(s,a)\end{aligned} \tag{10}$$

式中：δ_i为 TD 误差，δ_i的值越大，说明该条经验的误差值越大，算法更需要学习；$\lambda\left|\nabla_{\mathrm{A}}Q(s_t,a_t|\theta^{Q_a})\right|^n$为 A 网络中的动作损失影响程度，$n$是一个常数，本研究在算法设计中取$n=2$；参数$\varepsilon$的作用是防止采样时的经验概率为 0 的情况。

结合随机优先级和重要性采样方法，解决了以往 PER 算法模型存在性能下降和优先级存在导致的多样性损失问题。相比经典 PER 算法，优化过后的 PER 算法更具高效性。

2.3　专家纠偏策略

本研究基于无地图避障导航所使用的传感器是 2D 激光雷达传感器，机器人仅通过雷达传感器获取外界环境信息。虽然 TD3 导航避障算法在训练后期可以输出有效的运动指令引导坏机器人运动，但是在算法训练前期需要消耗较多时间输出有效的运动指令，严重影响了算法收敛速度。对此，本研究提出了一种新的导航避障策略——专家纠偏策略 (HCP)，该方法包括 2 个部分：危险评估模型、势场优化模型。

2.3.1　危险评估模型

为了将 2D 雷达传感器所获取的信息高效利用，根据雷达射线信息数据构建机器人的危险评估模型。通过 2D 雷达传感器得到的距离信息，首先定义危险因子f_{risk}，该危险因子为

$$f_{\mathrm{risk}}=\exp\left[\left(\frac{l_i}{l_{\max}}-k_1\right)^2-k_2\right] \tag{11}$$

式中：l_i为第i根雷达射线所测的距离信息，$l_{\max}$为表激光雷达传感器所能测得的最远距离值，k_1为危险因子的偏移量，k_2为偏置权重。k_1和k_2的作用是调整危险因子的平滑轨迹。再将危险因子值进行剪切与归一化，使得危险因子通过 2D 雷达传感器测得的距离取值范围为$f_{\mathrm{risk}}\in[0,1]$，如图 6。

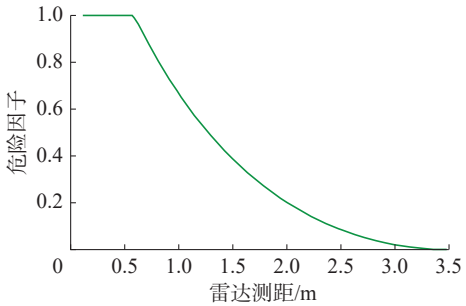

图 6　危险因子曲线

Fig. 6　Risk factor curve

随着机器人与障碍物之间的距离接近，危险因子数值逐渐增加。如果f_{risk}=1，说明在未知环境中机器人所处状态很危险，需要执行紧急避碰指令。该评估模型的建立与雷达传感器时避碰奖励值的设计类似，机器人在运动过程中危险值会随着雷达射线的测量值变化而变化，这与实际更加贴合。

2.3.2　势场优化模型

根据以上分析可知，当到达f_{risk}=1 的临界值时，定义了紧急避碰区域$d_{\min}$，机器人进入危险状态，如果深度强化学习算法输出的动作指令不能使机器人在避免与障碍物碰撞的同时接近目标，则下一时刻将根据 HCP 方法输出的动作命令来控制机器人的运动。针对雷达传感器所获取外界环境的信息，使用人工势场方法 (potential field planning, PFP) 对雷达射线的测量值处理，在机器人周围建立势场用于表征以机器人为中心的局部环境信息。

人工势场法的思想是建立吸引力势场和排斥力势场，如图 7 所示。在机器人导航避障过程中，目标对机器人有吸引力作用，环境中存在的障碍物对机器人存在排斥力作用，机器人在吸引力和排斥力的作用下躲避障碍物朝着合力方向移动前行。

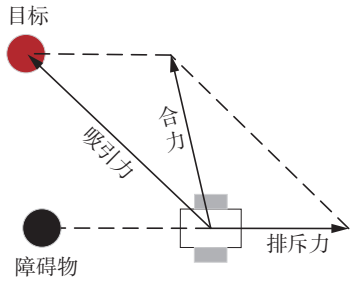

图 7　人工势场方法

Fig. 7　Artificial potential field method

人工势场包括 2 部分：由障碍物引起的排斥力势场和由目标位置形成的吸引力势场，公式为

$$\boldsymbol{U}_{\mathrm{PFP}}=\boldsymbol{U}_{\mathrm{att}}+\boldsymbol{U}_{\mathrm{rep}} \tag{12}$$

式中：$\boldsymbol{U}_{\mathrm{PFP}}$ 为合力势场，$\boldsymbol{U}_{\mathrm{att}}$ 为吸引力势场，$\boldsymbol{U}_{\mathrm{rep}}$ 为排斥力势场。其中吸引力势场和排斥力势场分别为

$$\boldsymbol{U}_{\mathrm{att}}=\frac{1}{2}\zeta(\boldsymbol{d}_{\mathrm{goal}}-\boldsymbol{d}_{\mathrm{UGV}})^{m} \tag{13}$$

$$\|\boldsymbol{U}_{\mathrm{rep}}\|=\begin{cases}\dfrac{1}{2}\eta\left(\dfrac{1}{l_i}-\dfrac{1}{d_{\min}}\right)^{n}, & l_i<d_{\min}\\ 0, \quad \text{其他}\end{cases} \tag{14}$$

式中：ζ 为引力势场因子，$\boldsymbol{d}_{\mathrm{goal}}$ 和 $\boldsymbol{d}_{\mathrm{UGV}}$ 分别为目标 $\left(X_{\mathrm{goal}}^{\mathrm{W}},Y_{\mathrm{goal}}^{\mathrm{W}}\right)$ 和机器人 $(X_{\mathrm{UGV}}^{\mathrm{W}},Y_{\mathrm{UGV}}^{\mathrm{W}})$ 在世界坐标系下的位置坐标，m 和 n 为可调参数，η 为斥力势场因子。

由于现实环境中障碍物较多，机器人受力十分复杂，传统人工势场函数计算的吸引力和排斥力会在某个位置存在大小相等、方向相反的情况，即机器人陷入局部最优。同时，在最开始机器人距离目标很远时，由于吸引力很大，极易忽略机体周围存在的较小排斥力，进而发生碰撞。

为解决上述问题，考虑到机器人要进入危险状态才能触发 HCP 方法，因此对排斥力势场公式进行优化，新的排斥力势场公式为

$$\|\boldsymbol{U}_{\mathrm{rep}}\|=\begin{cases}\dfrac{1}{2}\eta\left(\dfrac{1}{l_i}-\dfrac{1}{d_{\min}}\right)^{n}\left\|(\boldsymbol{d}_{\mathrm{goal}}-\boldsymbol{d}_{\mathrm{UGV}})\right\|^{k} & l_i<d_{\min}\\ 0, \quad \text{其他}\end{cases} \tag{15}$$

式(15)中增加一项 $\left\|\boldsymbol{d}_{\mathrm{goal}}-\boldsymbol{d}_{\mathrm{UGV}}\right\|^{k}$，作用是保证机器人在避免碰撞时的总势场值最小，即当机器人向目标移动并进入紧急避碰区域时，排斥力势场增大，但机器人与目标之间的距离减小，可以在一定程度上拖动斥力，解决局部最优问题。

为了得到由 HCP 方法输出的动作指令，需要通过势场函数计算机器人的旋转方向和旋转角度。由于移动机器人的角度计算是在机器人坐标系中进行的。因此，首先需要得到目标相对于机器人的位置，公式为

$$\begin{pmatrix}x_{\mathrm{goal}}\\ y_{\mathrm{goal}}\end{pmatrix}=\boldsymbol{R}^{\mathrm{T}}\begin{pmatrix}X_{\mathrm{goal}}^{\mathrm{W}}-X_{\mathrm{UGV}}^{\mathrm{W}}\\ Y_{\mathrm{goal}}^{\mathrm{W}}-Y_{\mathrm{UGV}}^{\mathrm{W}}\end{pmatrix} \tag{16}$$

式中：$(x_{\mathrm{goal}},y_{\mathrm{goal}})$ 为目标相对于机器人的位置，$\boldsymbol{R}^{\mathrm{T}}$ 为转换矩阵。由相对位置进一步计算目标相对于机器人的角度 θ_{goal}，公式为

$$\theta_{\mathrm{goal}}=\arctan\frac{y_{\mathrm{goal}}}{x_{\mathrm{goal}}} \tag{17}$$

获得机器人与目标的相对位置之后，根据吸引力势场和新的排斥力势场公式计算合力 F，用向量方式表示为

$$\boldsymbol{F}=\nabla\boldsymbol{U}_{\mathrm{att}}-\sum\nabla\boldsymbol{U}_{\mathrm{rep}} \tag{18}$$

式中：$\nabla\boldsymbol{U}_{\mathrm{att}}$ 为吸引力；$\sum\nabla\boldsymbol{U}_{\mathrm{rep}}$ 为机器人进入危险区域时，所有雷达射线所测距离小于 $d_{\min}$ 的排斥力累计值。由于本研究在仿真环境中使用的雷达射线有 24 根，所以在计算满足条件的雷达射线排斥力方向时的公式为

$$\theta_{l_i}=(i-12)\times 7.5 \tag{19}$$

式中 θ_{l_i} 为第 i 根雷达射线与小车前进方向的偏角。根据合力角度，可以计算出在机器人坐标系下排斥力在 x 方向和 y 方向的分力大小 F_x 和 F_y，通过三角函数计算后再求和。而合力角度通过所求分力得到，公式为

$$\theta_{\mathrm{H}}=\arctan\frac{F_y}{F_x} \tag{20}$$

式中，θ_{H} 为 HCP 方法所得到的机器人下一时刻的运动角度值。而机器人速度指令需要对其进行归一化处理，并将其转化为弧度。对此，本研究设计了弧度转换函数，将角度值进行处理输出在区间 $[-1,1]$，公式为

$$\theta_{\mathrm{RH}}=\begin{cases}1-\mathrm{e}^{-\frac{(\theta_{\mathrm{H}}-\zeta)^2}{2\sigma^2}}, & \theta_{\mathrm{H}}>0\\ \mathrm{e}^{-\frac{(\theta_{\mathrm{H}}-\zeta)^2}{2\sigma^2}}-1, & \text{其他}\end{cases} \tag{21}$$

式中 θ_{RH} 为 θ_{H} 通过弧度转换函数获得的机器人运动方向弧度值，用图表示得到图 8。

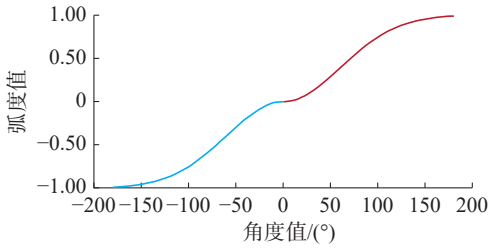

图 8　运动角度–弧度转化

Fig. 8　Transformation between angle and radian

在机器人到达危险区域，HCP 方法会输出上图的动作指令，而深度强化学习算法也会根据当前状态输出一个动作指令。为了更好选择机器人向目标移动的动作指令，需要比较 2 种动作指令的避障效果。通过将 2 个角度值分别输入到 C 网络中计算即时奖励值，选择高的即时奖励值所代

表的动作指令作为机器人需要执行的动作。

3　实验与结果分析

为验证 HCP-TD3 算法在机器人避障导航任务中的有效性和优越性。本研究分别在静态环境与动态环境下对所提出算法进行了对比与测试实验。共对比了 3 种算法，分别是：DDPG-PER 算法[26]、TD3 算法以及本研究提出的 HCP-TD3 算法。DDPG-PER 算法是 DDPG 算法[27] 与改进 PER 模型的结合，通过经验回放机制对 DDPG 算法训练过程中的经验进行存放、排序和抽取，以解决训练过程中的样本多样性损失问题。TD3 算法是指未使用改进优先经验回放策略和专家纠偏策略的基础 TD3 算法。

3.1　仿真环境及训练参数配置

本研究基于 ROS 系统，在 Gazebo 中构建搭载雷达传感器的机器人模型和导航避障模拟环境。机器人模型如图 9 所示，蓝色线条显示了机器人周围的激光雷达扫描光束。在仿真实验过程中，ROS 中的传感器节点获取激光雷达测量值，其视场为 180°，距离为 (0.12 m, 3.5 m)。实验采用 (−90°, 90°) 的 24 条激光雷达射线来收集机器人周围的数据。

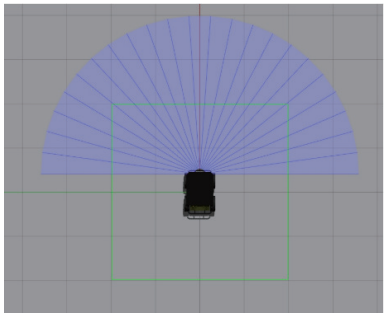

图 9　移动机器人模型
Fig. 9　Model of mobile robot

本研究在 Gazebo 中所搭建的导航避障模拟环境分为静态环境和动态环境，分别如图 10 和图 11 所示。

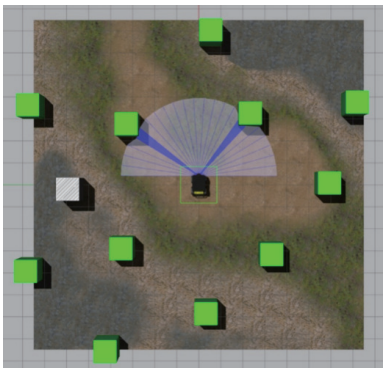

图 10　静态仿真环境
Fig. 10　Static simulation environment

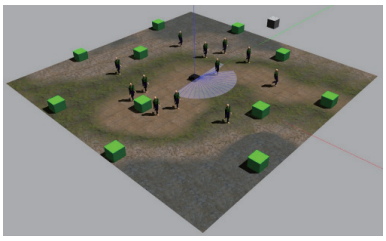

图 11　动态仿真环境
Fig. 11　Dynamic simulation environment

静态环境地图大小为 15 m×15 m，所有绿色方块都为静态障碍物，具有物理碰撞属性，白色箱体为目标物体，具有无重力属性，悬浮在地图上方。动态环境地图大小为 24 m×24 m，增加了 12 个随机运动的行人作为动态障碍物，每个行人运动速度各不相同，也更加符合真实情况。

本研究基于 PyTorch 和 TensorFlow 的深度强化学习框架对所需训练和对比的算法进行撰写。深度强化学习算法的训练参数选择将直接影响算法的收敛性，HCP-TD3 算法的主要训练参数见表 1。

表 1　HCP-TD3 算法的主要训练参数
Table 1　Main training parameters of HCP-TD3 algorithm

参数	训练回合	回合训练步数	记忆池大小	单次训练抽取样本数	奖励值衰减系数	噪声大小系数	A网络训练学习率	C网络训练学习率	C目标网络更新周期	A目标网络延时更新周期
具体数值	300	800	1×10^6	512	0.99	0.2	0.000 3	0.000 3	10	5

参考表 1 所设计的参数，DDPG-PER 算法不包含噪声和延迟更新，因此除了噪声大小系数和 A 目标网络延时更新周期以外，其余参数设计与表 1 中参数一致，TD3 算法则和表 1 中的参数值完全一致。

3.2　静态环境实验及结果分析

由图 10 可知，在训练阶段，机器人被放置在地图的中心，目标的位置随机初始化为环境的对角线区域。将 HCP-TD3 算法与其他算法在训练过程中的平均奖励值记录下来进行比较，如图 12 所示。

图 12 中的平均奖励值是由每连续 1 000 步的平均值进行计算画出的，以减少任务的随机性对奖励值的影响。计算这 3 种算法在 11 万步训练

过程中获得的平均奖励值，由图 12 可以看出，DDPG-PER 算法的平均奖励值曲线在大约 10 000 步的时候开始收敛，收敛区间为[4,6]；TD3 算法的平均奖励值曲线在大约 40 000 步的时候开始收敛，收敛区间为[6,8]；相比之下，本研究提出的平均奖励值以较慢的增长速度持续增长超过其他 2 种算法，在大约 60 000 步时开始收敛，收敛区间达到[8,10]。总体而言，本研究提出的算法的平均奖励值最大，做出的决策更好。

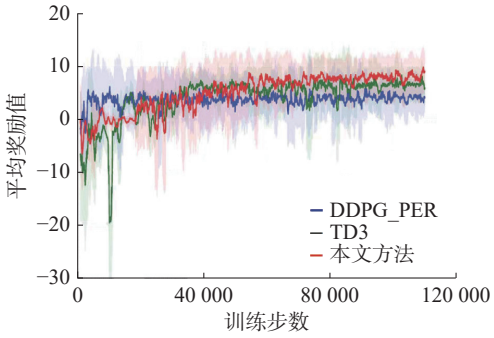

图 12　不同算法的奖励值对比

Fig. 12　Reward comparison of the proposed method with other methods.

运用 3 种算法训练完成得到的神经网络模型，通过额外的 100 个测试回合对所有的算法进行评估测试。首先定义了 3 个性能指标用于评估算法性能的好坏，即：机器人完成任务的成功率、完成任务的时间以及成功完成任务时的轨迹效率。任务成功率是指给定回合内机器人抵达目标区域的比率；任务完成时间是指机器人完成的所有任务平均时间；轨迹效率是指机器人完成的所有任务起点和终点之间的欧里几得距离平均值。

通过了 100 回合的算法评估测试，得到了 3 种算法下机器人避障性能的数据，具体数值见表 2。

表 2　静态环境下的算法性能数据对比

Table 2　Comparison of algorithm performance data in static environment

算法	成功率/%	完成时间/s	轨迹效率/m
DDPG-PER	25	73.55±1.20	14.64±0.75
TD3	79	64.36±10.29	11.11±1.52
HCP-TD3	93	60.70±8.78	10.25±1.48

在机器人成功率方面，本研究所提出的方法成功率最高（93%），而 DDPG-PER 和 TD3 算法的成功率较低，分别为 25% 和 79%。在完成时间方面，本研究的方法（60.70 s±8.78 s）与 TD3（64.36 s±10.29 s）和 DDPG-PER（73.55 s±1.20 s）的完成时间相近。在轨迹效率方面，本研究的方法平均值（10.25 m±1.48 m）低于 TD3(11.11 m±1.52 m) 和 DDPG-PER（14.64 m±0.75 m）。结果表明，该方法能在保证安全避碰的前提下保证最短的行驶路径和最高的成功率。

3.3　动态环境实验及结果分析

相比于静态环境，动态环境更为复杂，且障碍物运动轨迹难以预料。为评估本研究所提出 HCP-TD3 方法在动态环境下的导航避障性能，实验分别在动态仿真环境、虚拟工厂环境和现实环境中进行算法的对比和测试。

3.3.1　动态仿真环境评估

相比于静态仿真环境地图，动态仿真环境地图面积更加大、障碍物种类不同，环境更加复杂。将训练完成的深度强化学习导航避障模型用于动态环境中进行仿真测试，对比不同算法的性能。算法评估设置及过程与静态环境相同，用 100 回合内的 3 个性能指标对算法进行评估，得到了 3 种算法的机器人避障性能数据，见表 3。

表 3　动态环境下的算法性能数据对比

Table 3　Comparison of algorithm performance data in dynamic environment

算法	成功率/%	完成时间/s	轨迹效率/m
DDPG-PER	18	198.44±10.35	30.44±9.34
TD3	36	192.85±21.73	25.35±2.61
HCP-TD3	74	112.33±4.72	19.15±1.65

由表 3 可知，本研究的方法成功率达到 74%，明显高于 DDPG-PER（18%）和 TD3（36%）。在到达目标区域的完成时间方面，本研究的方法（112.33 s±4.72 s）比使用 DDPG-PER（192.85 s±21.73 s）和 TD3（192.85 s±21.73 s）方法节省了大量时间。在轨迹效率方面，本研究方法的平均轨迹（19.15 m±1.65 m）明显低于 DDPG-PER（30.44 m±9.34 m）和 TD3（25.35 m±2.61 m）方法。

3.3.2　虚拟工厂环境评估

为了进一步验证算法的可行性，本研究自主设计了工厂环境，如图 13 所示。该工厂环境设计的主要目的是模拟移动机器人在厂区内完成移动、搬运货物和躲避员工等任务，并且总体地图大小远远大于上一小节的动态环境地图。

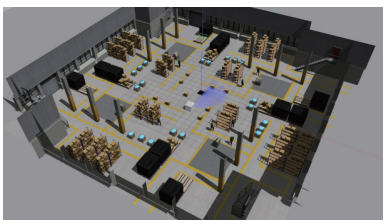

图 13　工厂环境

Fig. 13　Factory environment

图 13 中 4 个灰色方形区域为机器人初始位置区域，员工作为运动障碍物在货架和黄色标签内运动。算法测试时，移动机器人将随机出现在灰色区域，目标物体位置被设置为对角灰色方形区域。算法评估设置与先前动态环境一致，用 100 回合内的 3 个性能指标进行评估，具体数据见表 4。

表 4　工厂环境下的算法性能数据对比

Table 4　Comparison of algorithm performance data in factory environment

算法	成功率/%	完成时间/s	轨迹效率/m
DDPG-PER	32	121.81±1.32	42.88±3.02
TD3	48	104.72±1.71	36.73±2.54
HCP-TD3	82	103.34±0.72	33.18±1.92

本研究提出的方法成功率为 82%，而使用 DDPG-PER（32%）和 TD3（48%）的方法明显低于本研究方法。本研究方法所需的任务完成时间最短（103.34 s±0.72 s），而 DDPG-PER 方法需要 121.81 s±1.32 s，TD3 方法需要 103.34 s±0.72 s。在轨迹效率方面，本研究方法的平均距离（33.18 m±1.92 m）明显低于 DDPG-PER 方法（42.88 m±3.02 m）和 TD3 方法（36.73 m±2.54 m）。实验结果表明，模拟厂区环境下，HCP-TD3 算法控制的机器人可以在最短路径和最高成功率下完成任务。为了更加直观地对比算法性能数据，将其绘制成柱状图，如图 14 所示。

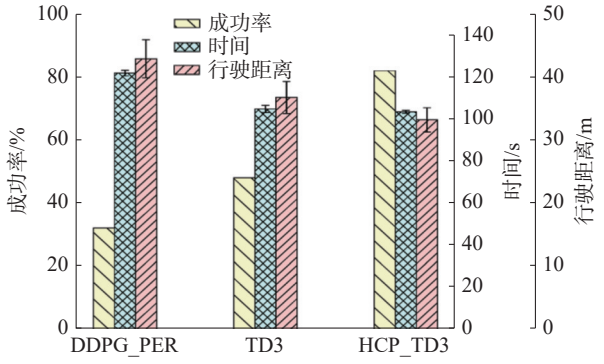

图 14　工厂环境算法性能对比

Fig. 14　Algorithm performance comparison in factory environment

3.3.3　现实环境实验测试

除了仿真环境外，本研究还将算法移植到真实机器人上进行真实环境测试。在现实环境中所使用的无人车移动平台，如图 15 所示。无人车底盘为驱控一体化智能移动底盘平台，主要包括移动动力组件（轮毂电机、车轮）、伺服驱动组件、电源组件和通讯模块组件，该移动平台具备差速控制 360°移动能力。除去底盘配置外，无人车移动平台还装配了 A2M6-R4 型号的高精度的 2D 激光雷达传感器，放置于工控机前方。

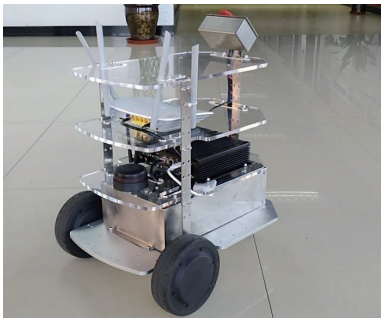

图 15　无人车移动平台

Fig. 15　Unmanned vehicle mobile platform

动态环境实验场地是一个室内环境，室内场地的大小为 12 m×12 m，走廊上有 4 个静态障碍物，每个障碍物的大小尺寸不一致，同时 2 个人作为移动障碍物在环境中随机移动。通过手动测量目标点与机器人的相对位置，并输入控制算法中，最终实验得到机器人动态环境避障导航结果，如图 16 所示。

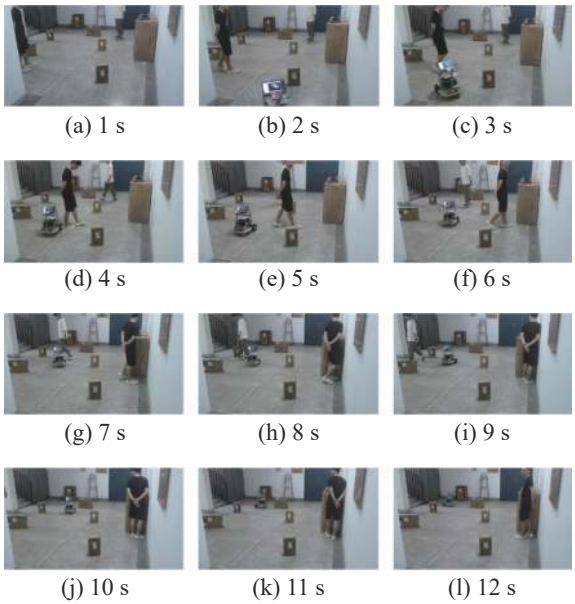

图 16　机器人动态环境测试结果

Fig. 16　Robot dynamic environment test results

机器人动态环境测试轨迹如图 17 所示，分析可知，在第 3 s 时，机器人检测到右侧静态障碍物和前方的动态障碍物；第 4~5 s，机器人转弯以避免碰撞；第 7 s，机器人检测到第 2 个动态障碍物，立即停止并原地转弯；在 12 s 内成功到达目标。实验证明，机器人能够在真实未知环境下，实现无地图自主导航避障。

图 17　机器人动态环境测试轨迹

Fig. 17　Robot dynamic environment test trajectory

总体而言，本研究提出的 HCP-TD3 算法不仅大大缩短了机器人到达目标区域的任务完成时间，而且在确保高成功率的同时还实现了更短的行走路径。该方法在实体机器人上取得了较好的应用效果，实现了动态环境下的避障导航。

4　结束语

本研究针对机器人无地图避障导航问题提出了一种基于 TD3 算法的改进方法，搭建了基于 ROS 的训练仿真环境，并将训练好的模型应用于实体机器人中。结果表明，采用 HCP-TD3 方法可以在保证避碰成功率和安全性的同时，获得最短的行驶轨迹和时间。但算法训练时间较长，且训练后期算法收敛速度较慢是此方法仍然存在的问题。未来可以将相机与激光雷达距离传感器相结合，对算法的收敛速度以及多传感器信息融合进行更多的研究，实现多传感器融合的复杂动态环境避障导航。

参考文献：

[1] DESOUZA G N, KAK A C. Vision for mobile robot navigation: a survey[J]. IEEE transactions on pattern analysis and machine intelligence, 2002, 24(2): 237−267.

[2] KRUSE T, PANDEY A K, ALAMI R, et al. Human-aware robot navigation: a survey[J]. Robotics and autonomous systems, 2013, 61(12): 1726−1743.

[3] ADAMKIEWICZ M, CHEN T, CACCAVALE A, et al. Vision-only robot navigation in a neural radiance world [J]. IEEE robotics and automation letters, 2022, 7(2): 4606−4613.

[4] ZHAO Haoning, WANG Chaoqun, GUO Rui, et al. Autonomous live working robot navigation with real-time detection and motion planning system on distribution line[J]. High voltage, 2022, 7(6): 1204−1216.

[5] EBADI K, BERNREITER L, BIGGIE H, et al. Present and future of SLAM in extreme environments: The DARPA SubT challenge[J]. IEEE transactions on robotics, 2024, 40: 936−959.

[6] SINGH K J, KAPOOR D S, THAKUR K, et al. Map making in social indoor environment through robot navigation using active SLAM[J]. IEEE access, 2022, 10: 134455−134465.

[7] GRANDO R B, DE JESUS J C, KICH V A, et al. Double critic deep reinforcement learning for mapless 3D navigation of unmanned aerial vehicles[J]. Journal of intelligent & robotic systems, 2022, 104(2): 29.

[8] LI Hanxiao, LUO Biao, SONG Wei, et al. Predictive hierarchical reinforcement learning for path-efficient mapless navigation with moving target[J]. Neural networks, 2023, 165: 677−688.

[9] LIKHACHEV M, FERGUSON D I, GORDON G J, et al. Anytime dynamic A*: An anytime, replanning algorithm[C]// ICAPS. Monterey, 2005, 5: 262−271.

[10] KHATIB O. Real-time obstacle avoidance for manipulators and mobile robots[M]//Cox IJ, Wilfong GT. Autonomous Robot Vehicles. New York: Springer, 1986: 396−404.

[11] NASIR J, ISLAM F, MALIK U, et al. RRT*-SMART: a rapid convergence implementation of RRT[J]. International journal of advanced robotic systems, 2013, 10(7): 299.

[12] DURRANT-WHYTE H, BAILEY T. Simultaneous localization and mapping: part I[J]. IEEE robotics & automation magazine, 2006, 13(2): 99−110.

[13] ALHMIEDAT T, MAREI A M, MESSOUDI W, et al. A SLAM-based localization and navigation system for social robots: the pepper robot case[J]. Machines, 2023, 11(2): 158.

[14] TAHERI H, XIA Zhaochun. SLAM: definition and evolution[J]. Engineering applications of artificial intelligence, 2021, 97: 104032.

[15] CHAPLOT D S, GANDHI D, GUPTA S, et al. Learning to explore using active neural SLAM[EB/OL]. (2020−04−10)[2021−01−01]. http://arxiv.org/abs/2004.05155.

[16] XUE Honghu, HEIN B, BAKR M, et al. Using deep reinforcement learning with automatic curriculum learning for mapless navigation in intralogistics[J]. Applied sciences, 2022, 12(6): 3153.

[17] HAN Yiheng, ZHAN I H, ZHAO Wang, et al. Deep reinforcement learning for robot collision avoidance with self-state-attention and sensor fusion[J]. IEEE robotics and automation letters, 2022, 7(3): 6886−6893.

[18] 欧阳勇平，魏长赟，蔡帛良. 动态环境下分布式异构多机器人避障方法研究[J]. 智能系统学报, 2022, 17(4): 752−763.
OUYANG Yongping, WEI Changyun, CAI Boliang. Collision avoidance approach for distributed heterogeneous multirobot systems in dynamic environments[J]. CAAI transactions on intelligent systems, 2022, 17(4): 752−763.

[19] ZHU Wei, HAYASHIBE M. A hierarchical deep rein-

forcement learning framework with high efficiency and generalization for fast and safe navigation[J]. IEEE transactions on industrial electronics, 2023, 70(5): 4962−4971.

[20] 李鹏, 阮晓钢, 朱晓庆, 等. 基于深度强化学习的区域化视觉导航方法[J]. 上海交通大学学报, 2021, 55(5): 575−585.
LI Peng, RUAN Xiaogang, ZHU Xiaoqing, et al. A regionalization vision navigation method based on deep reinforcement learning[J]. Journal of Shanghai Jiaotong University, 2021, 55(5): 575−585.

[21] PUTERMAN M L. Chapter 8 Markov decision processes [M]//Handbooks in Operations Research and Management Science. Amsterdam: Elsevier, 1990: 331−434.

[22] 刘克. 实用马尔可夫决策过程[M]. 北京: 清华大学出版社, 2004.
LIU Ke. Practical Markov decision processes[M]. Beijing: Tsinghua University Press, 2024.

[23] SILVER D, LEVER G, HEESS N, et al. Deterministic policy gradient algorithms[C]//International conference on machine learning. Beijing:ICML, 2014: 387−395.

[24] FUJIMOTO S, VAN HOOF H, MEGER D. Addressing function approximation error in actor-critic methods [EB/OL]. (2018−02−26)[2021−01−01]. http://arxiv.org/abs/1802.09477.

[25] SCHAUL T, QUAN J, ANTONOGLOU I, et al. Prioritized experience replay[EB/OL]. (2015−11−18)[2021−01−01]. http://arxiv.org/abs/1511.05952.

[26] LI Peng, DING Xiangcheng, SUN Hongfang, et al. Research on dynamic path planning of mobile robot based on improved DDPG algorithm[J]. Mobile information systems, 2021, 2021: 5169460.

[27] LILLICRAP T P, HUNT J J, PRITZEL A, et al. Continuous control with deep reinforcement learning[EB/OL]. (2015−09−09)[2021−01−01]. http://arxiv.org/abs/1509.02971.

作者简介:

田顺钰, 硕士研究生, 主要研究方向为智能自主无人系统。E-mail: shunyutian@163.com。

欧阳勇平, 硕士研究生, 主要研究方向为多机器人协作、智能无人系统。E-mail: oy15961483506@163.com。

魏长赟, 副教授, 博士, 主要研究方向为智能自主无人系统。发表学术论文 30 余篇。E-mail: c.wei@hhu.edu.cn。