文章编号:1671-4598(2025)11-0324-12 DOI:10.16526/j.cnki.11-4762/tp.2025.11.039 中图分类号:TP393.01 文献标识码:A

基于碰撞概率与速度障碍的深度强化学习安全导航研究

王军晓，王琨琨，陈豪驰
（浙江工业大学 信息工程学院，杭州 310023）

摘要：针对移动机器人在复杂多动态障碍物环境中导航效率低的问题，提出了一种基于碰撞概率和速度障碍的深度强化学习导航算法；为了保证导航策略的安全性，基于控制障碍函数设计了一个安全屏障用来调整动作；定义碰撞概率估计函数，用来评估障碍物发生碰撞的风险，将高风险的关键障碍物信息纳入深度强化学习算法的状态空间，减少特征提取时间；引入速度障碍理论设计了一种引导机器人针对关键障碍物进行主动避障的奖励函数，降低了机器人寻找最优航向角的时间；训练所得策略在不同环境下的测试结果验证该算法实现了安全快速导航。

关键词：机器人导航；深度强化学习；碰撞概率；速度障碍；控制障碍函数；安全屏障

Research on Deep Reinforcement Learning Safety Navigation Based on Collision Probability and Velocity Obstacles

WANG Junxiao, WANG Kunkun, CHEN Haochi
(College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China)

Abstract: To address the low navigation efficiency of mobile robots in a complex environment with multiple dynamic obstacles, a deep reinforcement learning navigation algorithm based on collision probability and velocity obstacles is proposed; To ensure the safety of navigation policies, a safety shield based on the control barrier function is designed to adjust actions. A collision probability estimation function is defined to evaluate the risk of collision with obstacles, and the critical obstacles information with high risks is incorporated into the state space of the deep reinforcement learning algorithm to reduce the time of feature extraction; A velocity obstacle theory is introduced to design a reward function to guide the robot to actively avoid critical obstacles, reducing the time for the robot to find the optimal heading angle. Through training strategy in different environments, the test results verify the safe and fast navigation of the algorithm.

Keywords: robot navigation; deep reinforcement learning; collision probability; velocity obstacles; control barrier function; safety shield

0 引言

机器人目前广泛应用于各行各业，在提高生产效率、解放人们于危险、恶劣和繁重工作、提高生活质量等方面展现了巨大的优势，而移动机器人则是机器人技术领域的一个重要分支，在工业、农业、服务业和医疗等领域取得了广泛应用[1-2]。具备自主感知、移动与决策能力的移动机器人正展现出令人惊叹的应用前景，将成为推动创新与发展的关键引擎之一[3-6]。

自主移动机器人由于其强大的运动规划能力和高度的自主性，逐渐应用于社会生活的各个方面，尤其是公共场所提供指导的服务型机器人[7]。如在医院环境中，移动机器人经常需要在走廊、病房和服务大厅之间来回穿梭，同时避免行人和医用推车等动态障碍物[8]；在银行、餐馆和车站等公共场所中，机器人需要在密集人群中保证安全性的同时快速完成位置引导、送餐和行李搬

收稿日期:2025-07-13； **修回日期**:2025-07-29。
基金项目:国家自然科学基金(62273306)。
作者简介:王军晓(1986-),男,博士,副教授。
通讯作者:王琨琨(1997-),男,硕士。
引用格式:王军晓,王琨琨,陈豪驰.基于碰撞概率与速度障碍的深度强化学习安全导航研究[J].计算机测量与控制,2025,33(11):324-335.

运等导航任务[9]。在这些环境中工作的移动机器人通常面临一个共同的问题，那就是快速变化的动态环境，特别是向各个方向移动的行人。当机器人在这种复杂多动态障碍物环境中进行导航任务时，不仅需要对人群和其他类型动态障碍物进行实时规划，还要考虑环境中静态障碍物的影响，这对自主移动机器人的导航要求大大提高[10]。在此背景下，深度强化学习提供了一种无需构建地图、直接基于局部感知信息进行决策控制的解决方案。端到端的深度强化学习模型能够实时感知并适应动态环境变化，完成未知和变化环境中的导航任务，从而推动移动机器人从执行任务到智能决策的演变[11]。

然而，当前大多数无地图导航研究都集中在静态环境，侧重于导航的成功率，对复杂多动态障碍物环境中导航的效率问题关注不足，同时缺乏避障的安全性考虑，而部分面向社会性环境的动态避障方法多以单步避障为目标，缺乏整体的导航能力[12]。

文献［13］使用深度神经网络与 Q-Learning 算法相结合的深度 Q 网络（DQN，deep q network）算法，实现了从图像的高维感知数据到机器人动作的端到端决策，但是 DQN 存在 Q 值过高估计和样本采样相关性的问题。对此，文献［14］使用 Double DQN 算法，采用两个神经网络分别进行动作选择和价值估计，有效解决了 Q 值高估问题，并训练室内环境中的移动机器人避开墙壁，实现了多种静态环境下的导航。文献［15］进一步利用优先经验回放机制，对样本进行优先性排序，高价值的样本可以被优先采样，有效解决了样本相关性问题，提高了训练效率，实现了 DQN 算法在真实场景中的高效导航。DQN 及其变体结构简单、应用广泛，但是仍存在一些问题。由于 Q-learning 是基于离散动作集合的，因此 DQN 算法不能处理连续动作空间，多用于状态空间维度较高但动作空间有限的导航任务。针对 DQN 在解决移动机器人连续动作空间存在的问题，文献［16］引入改进的深度确定性策略梯度（DDPG，deep deterministic policy gradient）算法，使用稀疏的高维激光雷达数据和目标点坐标作为输入，输出连续的转向动作，实现了机器人自主的端到端导航。文献［17］针对 DDPG 在较大环境中存在训练速度慢、稳定性差的问题，提出了一种使用全局路径规划算法生成多个子目标并结合 DDPG 进行大规模动态场景下导航的训练方法。文献［18］采用多步更新策略，引入双噪声机制同时结合人工势场法设计了一种综合奖励函数，有效提升了 DDPG 算法的训练效率和策略稳定性，实现了机器人在未知动态环境中的导航。虽然 DDPG 是一种较好的连续动作控制解决方法，常用于动作控制精度要求较高的导航任务，如转弯避障和动态场景，但是也存在 Q 值过估计、训练不稳定和收敛速度慢等问题。为此，文献［19］在 DDPG 的基础上提出了双延迟深度确定性策略梯度（TD3，twin delayed deep deterministic policy gradient）算法，采用双重评价网络估计 Q 值，并取较小的值作为训练目标，同时引入策略延迟更新机制，提高了策略学习稳定性，进一步在目标策略中加入噪声，避免 Q 值过拟合。文献［20］将概率路线图（PRM，probabilistic roadmap method）算法和 TD3 算法融合，实现了全局和局部的协同高效运动规划，在不同大小复杂室内环境中具有良好导航性能。文献［21］为了提高复杂动态环境中机器人导航的成功率，提出了一种结合动态窗口方法（DWA，dynamic window approach）和 TD3 算法的混合方法，实验结果表明 TD3-DWA 算法具有更高的可靠性，但是时间成本较高。文献［22］引入优先经验回放机制和长短期记忆（LSTM，long short-term memory）网络，提出了一种改进的 TD3 算法，解决了经典 TD3 算法收敛速度慢、样本效率低和无法捕捉时序依赖信息等问题，提高了多种室内环境下的导航效率。文献［23］针对复杂环境中移动机器人导航面临的死胡同问题，提出了一种避免死胡同且具有恢复能力的深度强化学习方法，减少了任务时间同时具有一定的安全保障。

虽然在提升导航效率和安全性方面已经有部分工作，但是在复杂多动态障碍物环境下的研究相对较少。因此，构建一种在复杂多动态障碍物环境下保证导航性能与避障安全性的深度强化学习导航方法，已成为实现移动机器人安全高效运行的关键问题之一。

1　问题建模

深度强化学习方法解决机器人导航问题的必要条件是导航过程具有完备的马尔科夫性质，即要求控制系统下一时刻决策只和当前时刻有关，而与历史时刻无关，并且能够据此建立马尔科夫模型。移动机器人执行导航任务过程中实时输入传感数据，输出运动控制命令，如果当前位置环境状态、速度、转向角已知，那么下一时刻机器人的环境状态和速度仅由前一时刻确定，而与历史时刻无关。因此，移动机器人导航满足完备的马尔科夫性，可以建模为一个马尔科夫决策过程（MDP，Markov decision process），将机器人和环境的交互过程映射为状态、动作、奖励等组成元素，然后使用强化学习方法求解最优策略。

将移动机器人在复杂多动态障碍物环境下的导航问题建模为一个 MDP，用五元组 $<S,A,P,R,\gamma>$ 来表示。其中 S 表示机器人在 t 时刻的全部状态信息：

$$S=(x_t,y_t,\theta_t,D_t,d_t,\alpha_t) \tag{1}$$

式中，(x_t,y_t) 表示机器人当前位置，θ_t 为当前时刻航向角，D_t 为传感器采集到的环境信息，d_t、α_t 分别为机器

人和目标的距离和相对角度。这种状态设计包含了完整的感知信息，使得导航策略能够在每一时刻仅依靠当前状态做出最优决策，保证系统的马尔科夫性质。A 表示每一时刻机器人可执行的动作集合，可以是离散的动作集合或连续的动作空间，对不同类型的移动机器人有不同的动作表示。动作描述了机器人根据状态做出的决策，决定了 MDP 的奖励和状态转移。R 表示 MDP 的即时奖励，表示当前状态下机器人执行动作 a_t 后获得的回报，用于指导策略优化，合理设计奖励函数可以提高学习效率与策略性能。P 表示在当前状态下采取动作后，系统转移到下一状态的概率分布，这一过程反映了机器人在当前位置和速度下，执行某个动作后，其位置、姿态和环境感知状态的变化。状态转移概率不仅和机器人自身的运动学、动力学模型有关，还受到环境动态变化和外部干扰的影响。所有可能状态的转移概率之和为：

$$\sum_{s_t} P(s_t, a_t, s_{t+1}) = 1 \quad \forall s_t \in S, \forall a_t \in A \tag{2}$$

式中，s_t 为 t 时刻状态，s_{t+1} 为可能转移的下一时刻状态，a_t 为机器人的动作。

机器人的目标是学到一个策略 π，即状态到动作的概率映射，使得在整个与环境交互的过程中可以获得最大的累积奖励，可以表示为：

$$R_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \tag{3}$$

式中，R_t 表示智能体在 t 时刻的累积奖励，$\gamma \in [0,1]$ 表示折扣因子。

深度强化学习方法通过优化基于神经网络的策略来最大化期望的折扣收益，是解决 MDP 问题最重要的工具之一，可以表示为：

$$L(\boldsymbol{\theta}) = E_{\pi_\theta}\left[\sum_{t=0}^{\infty} \gamma R^t\right] \tag{4}$$

式中，$L(\boldsymbol{\theta})$ 是总体的目标，$E_{\pi_\theta}[\cdot]$ 表示控制策略为 π_θ 时随机变量的期望值，γ 表示折扣因子。

2 导航算法设计

整体导航框架如图 1 所示，主要由以下 4 个模块组成：机器人和传感器、状态观测空间、神经网络和安全约束。机器人和传感器模块负责执行动作指令后由机器人从环境中获得原始传感器数据，包括移动机器人、雷达和惯性测量单元/里程计等硬件；状态观测空间对采集的数据进行处理得到神经网络的输入，包含激光雷达返回的测量数据 $\boldsymbol{o}_l^t$、目标点相关信息 $\boldsymbol{o}_g^t$、机器人位置速度信息 $\boldsymbol{o}_a^t$ 和关键障碍物信息 $\boldsymbol{o}_p^t$；神经网络模块使用不同的状态输入进行训练优化生成最优动作策略，包含所设计的网络结构；安全约束模块则对动作在满足所设定安全约束的条件下进行调整，包含一个基于控制障碍函数（CBF，control barrier function）的安全屏障。

2.1 状态空间设计

整个算法的状态空间主要由四部分组成，t 时刻机器人激光雷达的测量数据直接作为 $\boldsymbol{o}_l^t$，输入为 360 维数据；设定的目标点位置和计算得到的子目标点位置为 $\boldsymbol{o}_g^t$，输入为 4 维数据；机器人当前时刻的位置 (r_x, r_y) 和速度 (r_{vx}, r_{vy}) 为 $\boldsymbol{o}_a^t$，输入为 4 维数据；具有高风险的关键障碍物位置和速度信息，作为状态空间的 o_p^t 部分，其中 $i \in [1,2,\cdots,K]$，K 为关键障碍物数量。通过 360 维雷达数据，机器人可以获取周围障碍物的位置和距离信息，有助于机器人规划安全路径，维度大小要能够完整描述周围环境状态且不能太大，主要由雷达分辨率决定，对于高维激光雷达数据，可以进行降维处理；通过目标点相关信息，机器人能够实时调整最优路径；通过关键障碍物位置和速度信息，机器人可以感知周围动态障碍物的风险大小，进一步提高避障性能。因此，该状态空间设计具有合理性与必要性。

此外，为了确定关键障碍物个数 K，通过消融实验分别设置 $K=1$、4、8、12、16 在相同条件下进行训练

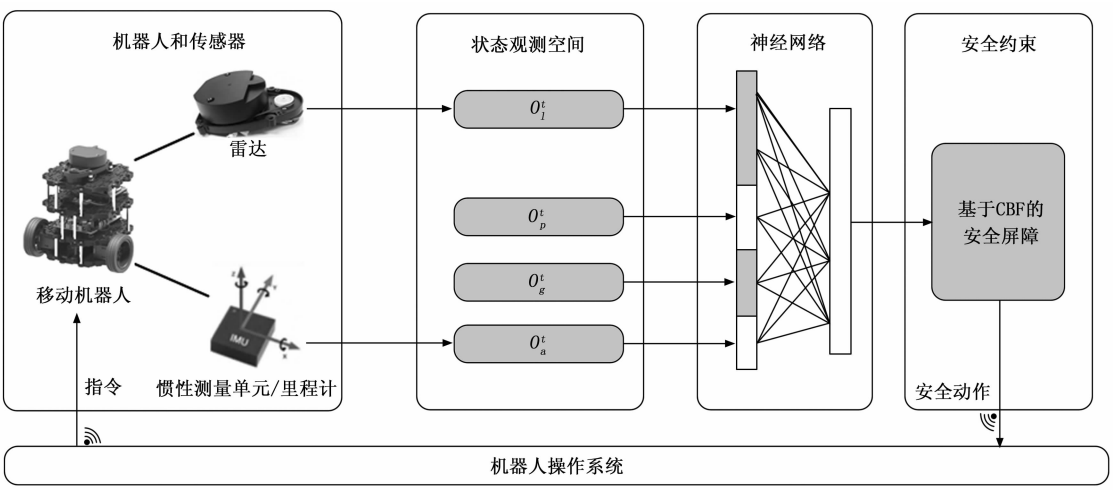

图 1 整体导航框架

测试，使用常见训练和测试指标评估不同 K 值对策略性能的影响。结果表明当 K 不断增大时，策略在训练开始可以获得更多障碍物相关的状态信息，提高了避障能力；但是状态空间维度也会上升，出现特征冗余和训练不稳定问题，干扰最优策略的学习，在导航测试中性能表现也相应较差。此外，不同 K 值下的训练损失函数曲线如图 2 所示，可知 $K=8$ 时收敛速度较快，且最终损失值较低，表明此时状态空间中包含的信息量和复杂度之间取得了较好的平衡，实现了更好的训练效果。综合考虑，最终选择 $K=8$ 作为所研究复杂多动态障碍物环境中的最优取值。实际应用时，可以根据环境中动态障碍物密度进行调整，当机器人扫描范围内没有 8 个障碍物时，用空集来表示。

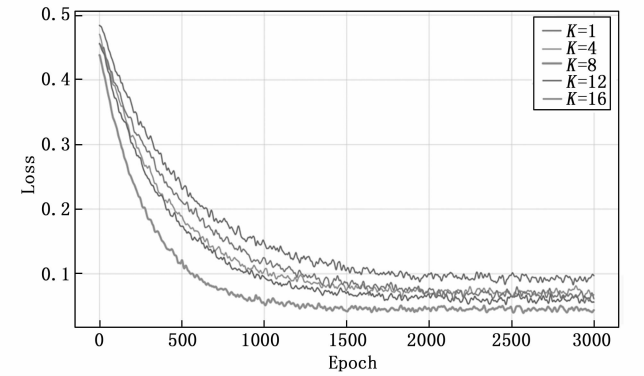

图 2　不同 K 值下的训练损失曲线

多种信息组合的观测空间可以表示为：

$$\boldsymbol{o}^t = [\boldsymbol{o}_l^t, \boldsymbol{o}_g^t, \boldsymbol{o}_a^t, \boldsymbol{o}_p^t] \tag{5}$$

为了获得状态空间设计中所需要用到的子目标点和关键障碍物信息，这里对相关设计原理作进一步说明。子目标点即连接机器人当前位置和目标位置与激光雷达最大测量距离的交点，分析可知当机器人进行了避障的情况下，就要更新子目标点，如果不更新，要先到原来的点再向目标点移动，即使机器人均为理想情况下的直线运动，总路径长度仍会增加，因此需要更新子目标点。

移动机器人和动态障碍物之间关系主要包括相对位置和相对速度，如图 3 所示，A 为机器人，B_1、B_2 表示障碍物，d_1、d_2 分别为机器人到两个障碍物的距离，$\boldsymbol{v}_A$、$\boldsymbol{v}_{B_1}$ 和 $\boldsymbol{v}_{B_2}$ 分别表示三者的速度，实线表示当前时刻 t，虚线为 $t+1$ 时刻。

由图可以看出，t 时刻 B_1 距离更近即 $d_1 < d_2$，而 B_2 的速度更快即 $|\boldsymbol{v}_{B_1}| < |\boldsymbol{v}_{B_2}|$，这样导致 $t+1$ 时刻机器人与 B_1、B_2 同时发生碰撞。因此，机器人周围动态障碍物的碰撞概率（CP，collision probability）表示为：

$$P = \alpha_1 P_d + \alpha_2 P_v \tag{6}$$

式中，P 为移动机器人和某一动态障碍物发生碰撞的概率；P_d 和 P_v 分别表示基于距离和速度的碰撞概率组成部分；α_1、$\alpha_2 \in [0,1]$ 为权重系数且 $\alpha_1 + \alpha_2 = 1$。

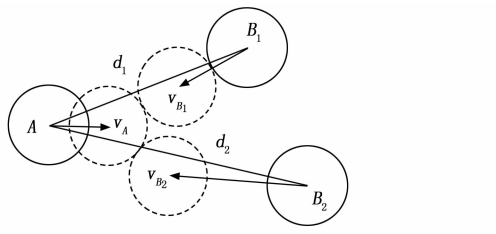

图 3　碰撞概率组成原理示意图

权重系数大小表示 CP 各组成部分的重要性，这里取 $\alpha_1 = \alpha_2 = 0.5$ 表示距离和速度对 CP 的影响相同，解释为同一时刻距离近、速度慢或距离远、速度快的障碍物 CP 是相等的。

P_d 计算公式如下：

$$P_d = \begin{cases} \dfrac{l_{\max} - \|d_o^t\|}{l_{\max} - l_{\min}}, & \text{if } \|d_o^t\| < l_{\max} \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

式中，$\|d_o^t\|$ 表示 t 时刻机器人和障碍物之间的距离，$l_{\max}$ 和 $l_{\min}$ 为激光雷达扫描的最大、最小距离。

P_v 计算公式如下：

$$P_v = \begin{cases} \dfrac{|v_A - v_B|}{|v_{A_{\max}}| + |v_{B_{\max}}|}, & \text{if } \alpha \in [\theta-\beta, \theta+\beta] \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

式中，v_A 和 v_B 分别表示移动机器人和障碍物的平移速度；同理 $|v_{A_{\max}}|$ 和 $|v_{B_{\max}}|$ 分别表示机器人和障碍物的最大平移速度；α 是相对速度 $v_{A,B}$ 的方向角；θ、β 是由图 3～4 所示的 VO 理论通过简单几何关系计算得到的角度常量。

CP 计算结果如图 4 所示，左边为仿真环境中某一时刻的状态，右边为当前时刻扫描范围内动态障碍物的碰撞概率计算结果。由图中可以看出，机器人左侧距离较近障碍物的碰撞概率为 31.2%，小于上方距离更远障碍物的 62.7%，说明此时上方障碍物的速度较快，进一步说明了所设计碰撞概率组成部分的合理性以及计算的可行性。

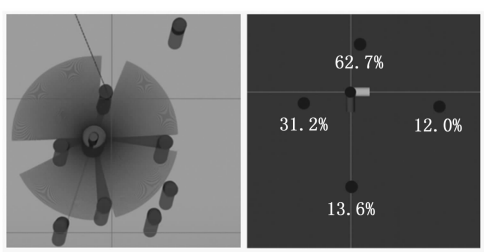

图 4　碰撞概率计算示意图

2.2　动作空间

动作空间由移动机器人的平移速度和旋转速度两部分组成，两者都定义在机器人的局部坐标系中，机器人

的前进方向定义为局部坐标 X 轴的正方向。因此，机器人的线速度始终在轴的正方向上，而逆时针旋转时角速度在正方向上。这里使用连续动作空间来获得更加准确的控制策略，动作空间的大小由实际硬件平台决定，因此机器人的动作空间可以定义为：

$$\boldsymbol{a}^t = [\upsilon_x^t, \omega_z^t] \tag{9}$$

式中，$\boldsymbol{a}^t$ 为深度强化学习模型生成的机器人动作，$\upsilon_x^t \in [0, 0.22]$ m/s、$\omega_z^t \in [-2, 2]$ rad/s 分别表示移动机器人局部坐标系下的平移速度和旋转速度。

2.3 奖励函数

深度强化学习方法一个基本问题是设计一个好的奖励函数，以引导智能体学习期望的行为。导航任务要求机器人能够快速到达目标的同时避免与静态或动态障碍物发生碰撞。因此，设计了一个多目标形式的奖励函数：

$$r = r_g + r_{\text{subg}} + r_c + r_a \tag{10}$$

式中，r 是引导机器人学习策略总的奖励函数，其他每一项的奖励函数形式描述如下。

1）r_g 是引导机器人到达最终目标点的奖励，是导航任务的首要目标，需要设计较大奖励值来确保成功到达目标在策略更新中具有主导作用。而机器人到达目标是一个稀疏事件，如果奖励过小，策略更新时信号不明显，学习速度慢，甚至无法学到有效策略。因此，最大奖励值要显著大于距离累积奖励，当机器人持续靠近目标却不达到时需要减小累积奖励系数，提高收敛速度和导航成功率，该项奖励定义为：

$$r_g = \begin{cases} r_{\text{goal}}, & \text{if} \quad \|d_g^t\| \leqslant g_{\text{error}} \\ -r_{\text{goal}}, & \text{else if } nstep \geqslant nstep_{\max} \\ r_{\text{dg}}(\|d_g^{t-1}\| - \|d_g^t\|), & \text{otherwise} \end{cases} \tag{11}$$

式中，$r_{\text{goal}} = 200$，$g_{\text{error}} = 0.2$ m，$nstep_{\max} = 500$，$r_{\text{dg}} = 32$。$\|d_g^t\|$ 表示 t 时刻机器人到目标点的距离；$\|d_g^{t-1}\|$ 表示 $t-1$ 时刻即上一步结束时机器人到目标点的距离；r_{goal} 为机器人到达目标所获得的奖励；r_{dg} 为引导机器人到达目标的奖励系数；g_{error} 表示到达目标所允许的误差范围；$nstep$ 表示 t 时刻本轮机器人所移动的总步数；$nstep_{\max}$ 表示回合步数的最大限制。当机器人到目标点的距离在误差允许范围内时，机器人获得最大奖励 r_{goal}，即认为机器人完成了导航目标；如果机器人在最大步数限制内没有到达目标，则认为导航失败，给予最大惩罚 $-r_{\text{goal}}$；其他情况下机器人随着每一步不断靠近目标而获得累积奖励。

2）r_{subg} 是到达子目标点的奖励项，子目标点即连接机器人当前位置和目标位置的直线与激光雷达当前最大扫描距离的交点。深度强化学习导航本身是一种局部规划方法，为了提高该类方法在远距离导航任务中的效率，可以将整个目标导航过程分为多个不同的子目标点导航任务，机器人到达子目标点也可以获得最大奖励。此外，要保证子目标点设计的合理性，机器人不能滥用子目标点奖励而忽略到达目标的奖励，该项奖励定义为：

$$r_{\text{subg}} = \begin{cases} r_{\text{goal}}, & \text{if} \|d_p^t\| \leqslant p_{\text{error}} \\ 0, & \text{otherwise} \end{cases} \tag{12}$$

式中，$p_{\text{error}} = 0.2$ m。p_{error} 表示到达子目标点所允许的误差范围；$\|d_p^t\|$ 表示 t 时刻机器人到子目标点的距离。当机器人到达子目标点误差范围内时就获得最大奖励值，其他时刻该项奖励值为零。

3）r_c 是用来惩罚机器人接近障碍物或发生碰撞的奖励项，碰撞也是机器人导航过程中最重要的情况之一，通常将其设为最大奖励的负值。此外，为了进一步避免碰撞发生，当机器人到障碍物的距离在一定范围内时可以给予惩罚，惩罚系数不能设置太大，避免机器人学习远离障碍物而不是避障的策略。该项奖励定义为：

$$r_c = \begin{cases} r_{\text{collision}}, & \text{if } \|d_o^t\| \leqslant d_{\text{robot}} \\ r_{\text{do}}(3d_{\text{robot}} - \|d_o^t\|), & \text{else if } \|d_o^t\| \leqslant 3d_{\text{robot}} \\ 0, & \text{otherwise} \end{cases} \tag{13}$$

式中，$r_{\text{collision}} = -200$；$d_{\text{robot}} = 0.12$ m；$r_{\text{do}} = -2$。$\|d_o^t\|$ 表示 t 时刻机器人到周围某个动态障碍物的距离，$r_{\text{collision}}$ 为发生碰撞给予的惩罚，d_{robot} 为判断碰撞发生的最大距离；r_{do} 为接近障碍物惩罚系数，表示当机器人靠近障碍物时获得惩罚。当机器人和障碍物之间的距离 $\|d_o^t\|$ 小于 d_{robot} 时，认为发生了碰撞；当位于 $d_{\text{robot}} \sim 3d_{\text{robot}}$ 之间时给予惩罚，越靠近障碍物惩罚值越大，其他情况下该项惩罚为零。

4）r_a 为考虑速度障碍（VO，velocity obstacles）与 CP 相结合的新型奖励项，用来引导机器人对关键障碍物进行主动避障，该项奖励系数设置需要在保证设计目标的基础上平衡奖励幅度，即不能太小导致可以忽略，又不能太大导致不稳定，造成路径波动大的同时增加时间成本，定义为：

$$r_a = r_{\text{cpvo}}(\theta_m - \theta_d^t) \tag{14}$$

式中，$r_{\text{cpvo}} = 5$ 是主动避障奖励系数；$\theta_m = \pi/4$ 是设定的最大转向角，即期望机器人每一次的转向角度不超过设定的最大值；θ_d^t 表示考虑关键障碍物时的机器人实际转向角度，这一项新的奖励主要是用来引导机器人的对转向角速度调整。为了寻找期望的方向角 θ_d^t，对速度障碍的概念进行扩展，如图 5 所示。

速度障碍 $VO_{A,B}$ 表示机器人 A 当前速度在未来某个时间与障碍物 B 发生碰撞的速度空间，为了描述 $VO_{A,B}$，首先定义一个特殊的占用区域：

$$SO_{A,B} = \{\boldsymbol{p}_S \mid d(\boldsymbol{p}_S, \boldsymbol{p}_B) < r_A + r_B\} \tag{15}$$

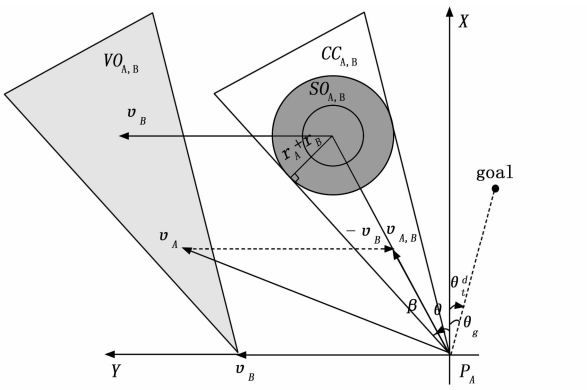

图 5　速度障碍理论扩展示意图

式中，$d(\boldsymbol{p}_S,\boldsymbol{p}_B)$ 表示 $\boldsymbol{p}_S$ 和 $\boldsymbol{p}_B$ 之间的距离。然后碰撞锥 $CC_{A,B}$ 可以定义为：

$$CC_{A,B}=\{\boldsymbol{v}_{A,B}\mid \exists\, \boldsymbol{t},\boldsymbol{v}_{A,B}\boldsymbol{t}\cap SO_{A,B}\neq\phi\} \tag{16}$$

最后，$VO_{A,B}$ 定义为：

$$VO_{A,B}=CC_{A,B}\oplus\boldsymbol{v}_B \tag{17}$$

式中，$\oplus$ 表示闵可夫斯基向量和运算符。

$CC_{A,B}$ 的物理意义是对于任意的相对速度 $\boldsymbol{v}_{A,B}\in CC_{A,B}$ 都会使机器人和障碍物在将来的某个时间发生碰撞。从相对速度的方向角来看，$CC_{A,B}$ 也可以定义为：

$$CC_{A,B}\in[\theta-\beta,\theta+\beta] \tag{18}$$

式中，θ、β 的计算公式可以表示为：

$$\theta=\arctan2\left(\frac{\boldsymbol{p}_{B_y}}{\boldsymbol{p}_{B_x}}\right),\beta=\arcsin\left(\frac{\boldsymbol{r}_A+\boldsymbol{r}_B}{\|\boldsymbol{p}_B\|}\right) \tag{19}$$

通过跟踪 t 时刻机器人周围具有高碰撞概率的关键障碍物，并利用碰撞锥和子目标点的信息，使用基于采样的搜索算法，机器人可以找到导航到子目标点的最优方向角 θ_d^t。

2.4　深度强化学习算法

这里使用的 DDPG 算法的改进，双延迟深度确定性策略梯度（TD3）深度强化学习（DRL，deep reinforcement learning）算法，是一种用于连续动作控制的，基于策略函数（Actor）和价值函数（Critic）的算法。TD3 算法主要使用了三个关键技术：双 Q 网络、延迟策略更新和目标策略加入噪声。双 Q 网络通过取两个 Q 值中的最小值来抑制动作价值的过高估计，提升了策略的稳定性；延迟策略更新和目标策略加入噪声则有助于缓解策略过拟合问题，增强策略的探索能力和适应性。

算法 θ_d^t 伪代码如下所示：

算法：寻找期望的方向角

输入：目标方向角 θ_g，关键障碍物 $\boldsymbol{K}$，移动机器人线速度 $\boldsymbol{v}_{A_x}$，采样数量 N

输出：最优的方向角 θ_t^d

1）初始化：$\theta_t^d\leftarrow\frac{\pi}{2}$

2）if $K\neq\phi$ then

3）$\theta_{\min}\leftarrow\infty$

4）for $i=1,2,\cdots,N$ do

5）$\theta_u\leftarrow$ sample from $[-\pi,\pi]$

6）free←True

7）for k in K do

8）$\theta_{v_{A,B}}\leftarrow\text{atan 2}\left(\frac{\boldsymbol{v}_{A_x}\sin(\theta_u)-\boldsymbol{v}_{B_y}}{\boldsymbol{v}_{A_x}\cos(\theta_u)-\boldsymbol{v}_{B_x}}\right)$

9）$\theta,\beta\leftarrow$ from（求）using k

10）if $\theta_{v_{A,B}}\in[\theta-\beta,\theta+\beta]$ then

11）free←False

12）break

13）if free then

14）if $\|\theta_u-\theta_g\|<\theta_{\min}$ then

15）$\theta_{\min}\leftarrow\|\theta_u-\theta_g\|$

16）$\theta_t^d\leftarrow\theta_u$

17）else

18）$\theta_t^d\leftarrow\theta_g$

19）return θ_t^d

2.5　安全屏障

带有安全屏障的导航框架如图 6 所示，其中 DRL－CPVO 为所设计的基于 CP 和 VO 的深度强化学习导航算法，整体结构与强化学习交互模型相同，这里只对 DRL 生成的动作 u_{DRL} 使用 CBF 安全屏障进行约束得到优化调整后的动作 $u_{\text{DRL}}+u_{\text{CBF}}$，$u_{\text{CBF}}$ 为补偿的动作。因此，可以把整个导航算法称之为带有安全屏障的深度强化学习导航算法，即 DRL-CPVO-CBFs 算法。

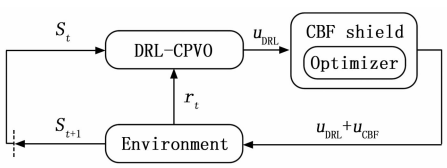

图 6　安全屏障导航框架示意图

为了设计安全屏障，考虑机器人的位置是受约束的状态，因此安全集合可以表示为：

$$C=\{x\in\frown\mathbb{R}^2\mid(x_r-x_o)^2+(y_r-y_o)^2\geqslant d_{\text{safe}}^2\} \tag{20}$$

式中，x_r、y_r 表示机器人的位置，x_o、y_o 表示障碍物的位置，d_{safe} 为定义的安全距离：

$$d_{\text{safe}}=d_0+\lambda P_{\max} \tag{21}$$

式中，d_0 表示基本的安全距离，λ 是考虑碰撞概率项对安全距离的影响，$P_{\max}$ 表示关键障碍物的碰撞概率。将碰撞概率引入距离约束设计，可以使机器人在面临多个不同风险障碍物的情况下找到最合理的 d_{safe}，即当机器人和障碍物距离较远但碰撞风险大时，为保证安全也应该增加 d_{safe}，从而可以得到一个动态变化的安全关键距离约束。这里取 $d_0=\lambda=0.2$，表示最危险情况下即 $P=1$ 时需要考虑 2 倍的安全距离进行限制，参数的取值

主要根据机器人和障碍物的最大速度进行调整。

两轮差速移动机器人的运动学方程可以描述为如下形式：

$$\begin{cases}\dot{x}_r = v_r\cos\theta_r \\ \dot{y}_r = v_r\sin\theta_r \\ \dot{\theta}_r = \omega_r\end{cases} \tag{22}$$

式中，v_r 为机器人局部坐标系下的线速度，ω_r 为角速度。将运动学模型离散化可得：

$$\begin{cases}x_{t+1} = x_t + v_t\cos\theta_t \cdot \Delta t \\ y_{t+1} = y_t + v_t\sin\theta_t \cdot \Delta t \\ \theta_{t+1} = \theta_t + \omega_t \cdot \Delta t\end{cases} \tag{23}$$

式中，x_{t+1} 、y_{t+1} 表示 $t+1$ 时刻机器人在平面中的位置，θ_{t+1} 表示 $t+1$ 时刻的朝向角；x_t 、y_t 表示 t 时刻机器人在平面中的位置，θ_t 表示 t 时刻的朝向角；v_t 和 ω_t 分别表示 t 时刻机器人的线速度和角速度；Δt 表示时间步长。

令 x_t 、x_{t+1} 分别表示机器人在 t 时刻和 $t+1$ 时刻的状态，即：

$$x_t = \begin{bmatrix}x_t \\ y_t \\ \theta_t\end{bmatrix}, x_{t+1} = \begin{bmatrix}x_{t+1} \\ y_{t+1} \\ \theta_{t+1}\end{bmatrix} \tag{24}$$

则机器人在控制输入作用下的离散时间状态更新模型可以表示为：

$$x_{t+1} = f(x_t, u_t) \tag{25}$$

式中，$f(x)$ 为机器人系统的运动学状态转移函数，$u_t = [v_t, \omega_t]$ 表示 t 时刻机器人的控制输入。

对于上述的离散时间状态更新模型，假设存在一个由连续可微函数 $h:\mathbb{R}^n \to \mathbb{R}$ 定义的安全集合 C 表示成如下形式：

$$C:\{x \in \mathbb{R}^n : h(x) \geqslant 0\} \tag{26}$$

式中，$C \subset D \subset \mathbb{R}^n$ ，D 为系统所定义的可行状态集合。当系统的状态 x 能够一直保持在 C 内时，集合 C 是前向不变的。如果存在 $\eta \in [0,1]$ 使得：

$$\forall x_t \in D \quad s.t.\, h(x_{t+1}) \geqslant (1-\eta)h(x_t) \tag{27}$$

式中，η 反映了控制障碍函数推动系统状态向安全集合 C 内收敛的程度，η 越小推动作用越强。

上式是对系统状态的约束，如果要获得对机器人动作形式的 CBF 约束，将式（25）代入可得：

$$\forall x_t \in D \ \exists u_t\, s.t.\, h[f(x_t, u_t)] + (\eta-1)h(x_t) \geqslant 0 \tag{28}$$

因此，关于机器人状态的 CBF 函数可以设计为如下形式：

$$h(x_t) = (x_t - x_o)^2 + (y_t - y_o)^2 - d_{\text{safe}}^2 \tag{29}$$

由式（28）、（29）相结合可得：

$$(x_{t+1} - x_o)^2 + (y_{t+1} - y_o)^2 - d_{\text{safe}}^2 + (\eta-1)h(x_t) \geqslant 0 \tag{30}$$

将式（23）可得代入上式可得：

$$[(x_t + v_t\cos\theta_t \cdot \Delta t - x_o)^2 + (y_t + v_t\sin\theta_t \cdot \Delta t - y_o)^2] - d_{\text{safe}}^2 + (\eta-1)h(x_t) \geqslant 0 \tag{31}$$

整理成标准形式可得：

$$Av_t^2 + Bv_t + C \geqslant 0 \tag{32}$$

式中，$A = (\cos^2\theta_t + \sin^2\theta_t) \cdot \Delta t^2 = \Delta t^2$ ；$B = 2\Delta t[(x_t - x_o)\cos\theta_t + (y_t - y_o)\sin\theta_t]$；$C = (x_t - x_o)^2 + (y_t - y_o)^2 - d_{\text{safe}}^2 - (1-\eta)h(x_t)$ 。

为了将 CBF 安全约束集成到优化问题中，即将强化学习导航模型输出的机器人动作，通过一定程度的调整，投影为安全的动作，以达到 CBF 安全屏障的效果。本着最小干涉的原则，调整后的安全动作应尽量接近原动作，非必要不干涉和修改原动作。也就是说，如果原输出动作被判定是安全的，那么不对其进行调整；反之，如果是不安全的，就找到一个最小的调整量，使得最终输出的动作是和原始动作最为接近的安全动作。因此，该优化问题可以表示为：

$$\begin{aligned} u_t^* &= \text{argmin}_{u_t \in A} \left\| u_t - u_{\text{DRL}} \right\|_2^2 \\ &s.t.\, Av_t^2 + Bv_t + C \geqslant 0 \end{aligned} \tag{33}$$

式中，$A = \{u_t \mid v_{\min} \leqslant v_t \leqslant v_{\max}, \omega_{\min} \leqslant \omega_t \leqslant \omega_{\max}\}$ 为机器人的动作空间。u_t^* 为最终修正后得到的满足约束条件的安全动作，u_{DRL} 表示深度强化学习导航策略生成的动作。当该优化问题没有解时，机器人不执行动作，防止可能发生碰撞问题，并将该回合记为失败。

由于只有凸优化问题可以使用 QP 求解器在线快速求解，因此在求解该优化问题时，要先对 CBF 约束分析。式（32）指出 $A > 0$ ，由一元二次不等式相关知识可知当 $B^2 - 4AC > 0$ 时该约束条件存在两个不连续的解区间，此时分别对两个区间求解凸二次优化问题，取最优解中代价最小的一个作为最终动作；其他情况下可以直接求解该优化问题。此外，由于角速度仅受边界约束，不参与优化计算。根据内点法的时间复杂度估计该问题计算复杂度为 $O(1)$ ，单次 QP 求解时间成本较低。因此，该方法具有良好的实时性与可部署性。

3 算法训练和测试

3.1 训练设置

为了训练所提算法的导航策略，搭建了如图 7 所示的 Gazebo 仿真环境，环境的大小为 16 m^2，蓝色圆柱体用来表示半径为 0.05 m、高度为 0.2 m 的障碍物，这些障碍物被随机放置在环境中的不同位置，同时可以向不同的方向以 [0，0.2] m/s 的随机速度移动，这表明机器人能够与环境中的任何一个障碍物发生碰撞。

机器人的起始位置为（−1.5，1.5），目标位置为（1.5，1.5），图中的右上角方形区域是允许的目标误差范围，用于衡量机器人是否成功到达目标点。每回合训练开始时，机器人都从起始位置出发向目标点移动，当

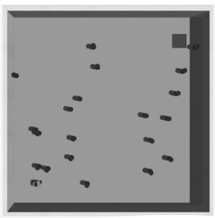

图 7　算法训练环境

成功到达目标、发生碰撞或者超过最大步数限制时，该回合训练结束，并重置环境，机器人和障碍物回到初始位置。然后，为了更加体现算法设计本身的性能差异，使用相同的参数设置分别训练三种算法模型：DRL 算法模型、加入 VO 奖励项的 DRL-VO 算法模型以及考虑 CP 和 VO 奖励项并且带有安全屏障的 DRL-CPVO-CBFs 算法模型。此外，训练使用的是 TD3 强化学习算法，相关参数设置如表 1 所示。

表 1　训练相关参数设置

超参数	设定值
策略网络学习率	0.000 1
评价网络学习率	0.001
经验回放缓冲区大小	1 000 000
折扣因子	0.99
软更新系数	0.005
延迟更新步数	2
策略噪声标准差	0.2
噪声截断范围	0.5
动作高斯噪声	0.1
每回合最大步数	500
总回合数	3 000

3.2　仿真环境测试

为了对不同的测试环境进行描述，首先定义了一个动态障碍物密度的概念：

$$\rho_d = \frac{N}{S} \tag{34}$$

式中，ρ_d 表示环境中的动态障碍物密度，N 表示环境中动态障碍物的数量，S 表示环境的面积大小。ρ_d 可以理解为环境中单位面积内动态障碍物的数量，直接体现了多动态障碍物下环境的复杂程度，使得环境的描述更加准确，方便算法在不同环境下性能的对比分析。

此外，为了体现不同算法的导航性能，使用了如下常用的 4 个性能指标加上一个用于衡量带有安全屏障算法性能的安全得分指标。

1）成功率：机器人在所有测试回合中成功到达目标并且没有发生碰撞的比例；

2）时间成本：机器人成功到达目标的平均时间，即所有测试回合中不发生碰撞情况下到达目标所用时间成本的平均值；

3）平均长度：机器人成功到达目标的平均路径长度，即所有测试回合中不发生碰撞情况下多次到达目标所走过的平均路径；

4）平均速度：平均路径与时间成本的比值，即所有测试回合中成功到达目标所用速度的平均值。

5）安全得分：安全得分可以定义如下形式：

$$\text{安全得分} = (1 - \frac{k}{N}) \times 100 \tag{35}$$

式中，k 表示机器人每回合测试中发生安全违规的次数，N 表示移动的总步数。安全违规就是当机器人和动态障碍物之间的距离小于机器人的自我半径时，就认为发生了一次安全违规，这里将机器人的自我半径设置为 0.2 m。

通过 ROS 实时订阅机器人的速度和位置数据并发布，将测试过程中每一回合的数据都记录下来求平均值。

为了对训练得到的策略进行验证，在和训练环境相同大小的环境中分别设置了 6 组不同 ρ_d 仿真环境下的对比实验，在不考虑动态障碍物数量的情况下，将训练环境称为环境 I，如图 8 所示。

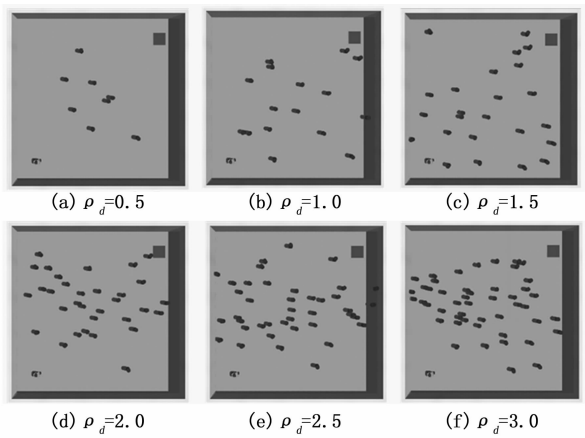

图 8　环境 I 中不同数量的动态障碍物

每次在不同的环境中分别使用①DWA、②DRL、③DRL-VO、④DRL-CPVO-CBFs 四种算法进行 100 回合测试，除成功率外所有性能指标均为 100 回合的平均值，结果如表 2 所示。

由表 2 中结果可以看出：经典 DWA 算法的成功率略高于基本的 DRL 算法，但是都小于另外两种 DRL 方法，虽然在所有环境中都有着较短的平均路径长度但是所用时间成本也相对较高，因而导航效率较低，且安全性不足。带有安全屏障的 DRL-CPVO-CBFs 算法在不同动态密度下都有较好的导航性能。首先，在导航成功率方面，即使是在动态密度最高的环境中，所提算法仍然有着 0.85 的成功率，比基本的 DRL 算法提高了 39.3%，比 DRL-VO 算法也有 13.3%的提升，表明该算法安全性能进一步得到了提升，这点由安全得分表现也可以看出来，说明机器人在成功到达目标的过程中发

表 2 环境 I 中的仿真测试结果

环境 I	算法	成功率/%	时间成本/s	平均路径/m	安全得分
$\rho_d = 0.5$	①	93	22.86	4.36	98.5
	②	85	25.45	5.09	98.5
	③	96	22.10	4.42	99.2
	④	96	21.95	4.39	99.6
$\rho_d = 1.0$	①	86	24.77	4.64	96.2
	②	82	27.65	5.53	97.3
	③	95	23.30	4.66	98.6
	④	96	23.40	4.68	99.5
$\rho_d = 1.5$	①	83	26.81	4.73	95.0
	②	81	29.74	5.65	97.1
	③	93	23.95	4.79	98.3
	④	95	22.62	4.75	99.4
$\rho_d = 2.0$	①	77	32.90	4.80	93.5
	②	75	33.61	6.05	95.8
	③	89	24.85	4.97	97.3
	④	93	24.25	4.85	99.2
$\rho_d = 2.5$	①	69	33.52	5.00	92.3
	②	66	32.53	6.18	94.1
	③	82	25.90	5.18	96.7
	④	88	24.80	4.96	98.8
$\rho_d = 3.0$	①	63	34.12	5.05	90.2
	②	61	30.90	6.18	91.7
	③	75	27.15	5.43	95.9
	④	85	25.05	5.01	98.3

生安全违规的次数非常少，所设计的安全屏障发挥了作用。然后，从平均时间成本可以得出随着环境动态密度的增加，DRL-CPVO-CBFs 所用时间成本增加较少，导航效率比 DRL 提升了 18.9%，说明考虑 CP 和 VO 的深度强化学习导航算法使机器人学到了更优的策略。最后，所提算法路径长度也更加短，说明该导航算法控制的机器人运动更加稳定，进一步提升了导航效率。因此，DRL-CPVO-CBFs 算法实现了复杂多动态障碍物环境中的安全快速导航。

为了进一步验证所提导航算法的泛化性能，在训练环境基础上变化得到了三种新的环境：与训练环境大小相同(16 m^2)但是包含静态障碍物和其他类型动态障碍物的环境 II、III，与训练环境大小不同(25 m^2)且包含静态障碍物和其他类型动态障碍物的环境 IV。同时分别在环境 II、III、IV 中设置了两种不同动态密度的障碍物，如图 9 所示。

同样在不同的环境中使用不同的算法分别进行 100 回合测试，测试结果如表 3。由表中的测试结果可以得出以下结论：DWA 算法在不同环境中依然有着最短的平均路径长度，但是成功率、时间成本和安全得分等性能指标都有较大程度的下降，表明经典算法在更加复杂环境中导航稳定性较低。DRL-CPVO-CBFs 算法在更大更加复杂的多动态障碍物环境中依然有着良好的导航性

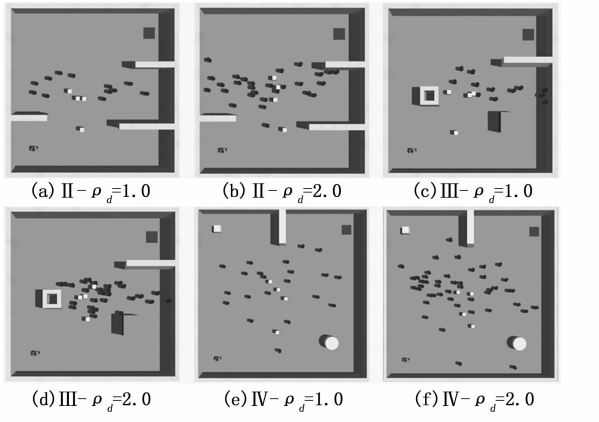

图 9 环境 II、III 和 IV 中不同数量的动态障碍物

能。首先，在所有不同的环境下所提算法都有着 0.85 以上的成功率，在 III－ρ_d＝2.0 环境中，相比 DRL 算法成功率提高了 63.5%，比 DRL-VO 算法高了 16.4%。其次，安全得分方面也普遍高于其他算法，进一步说明了所提算法的安全可靠性。最后，从时间成本可以看出导航效率比 DRL 算法最大提升了 21.9%，比 DRL-VO 最大提升了 16.6%。因此，测试结果表明 DRL-CPVO-CBFs 算法具有更强的适应性能。

表 3 环境 II、III、IV 中的仿真测试结果

环境	算法	成功率/%	时间成本/s	平均路径/m	安全得分
II－ρ_d＝1.0	①	81	32.88	5.29	95.1
	②	76	33.02	6.55	95.7
	③	89	28.13	5.94	96.5
	④	90	26.12	5.32	97.6
II－ρ_d＝2.0	①	65	36.75	5.56	93.0
	②	62	35.94	6.72	93.5
	③	80	33.08	6.00	95.2
	④	86	28.56	5.60	96.3
III－ρ_d＝1.0	①	77	33.97	5.48	96.9
	②	69	34.85	6.81	97.8
	③	83	32.42	6.25	98.4
	④	88	28.93	5.50	99.1
III－ρ_d＝2.0	①	56	41.05	5.86	94.5
	②	52	39.58	7.20	96.1
	③	73	37.06	6.77	97.2
	④	85	30.91	5.91	98.3
IV－ρ_d＝1.0	①	84	45.10	7.22	93.0
	②	76	45.46	9.08	93.5
	③	89	43.84	8.33	95.2
	④	93	38.21	7.26	96.7
IV－ρ_d＝2.0	①	68	50.66	7.57	88.6
	②	65	49.42	9.39	90.8
	③	84	47.50	8.55	92.4
	④	90	39.97	7.60	94.5

此外，选取了环境 I 中动态密度最大时，接近于测试性能指标平均值的回合，绘制了如图 10 所示三种算法不同指标的对比图。

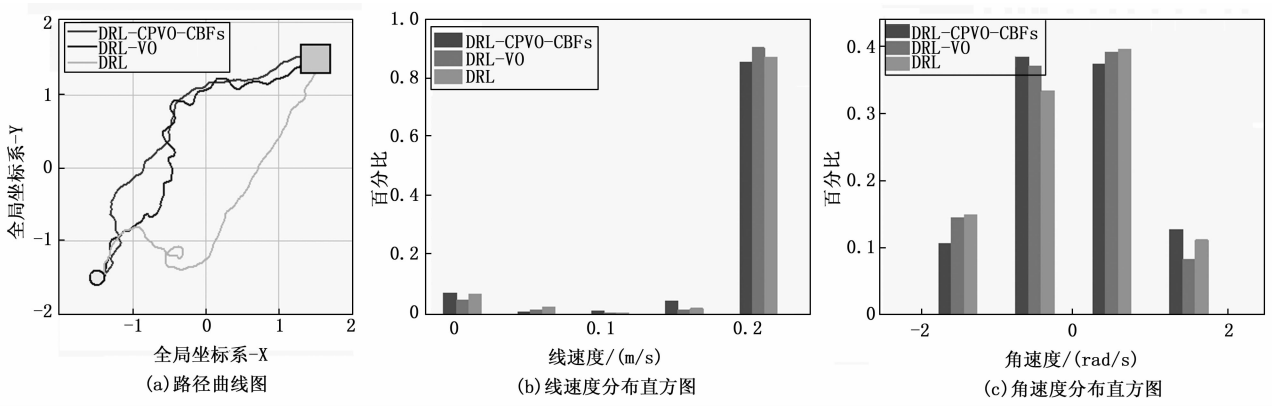

图 10　导航过程不同指标对比图

从相同环境下导航结果的路径曲线图可以看出，DRL-VO 和 DRL-CPVO-CBFs 生成的路径相对来说更加短，这说明基于 VO 的奖励模型学会了在多动态障碍物环境中寻找最优航向角的策略，但是 DRL-VO 规划的路径更加曲折，这表明机器人在实时导航过程中需要考虑所有跟踪障碍物来寻找最优的航向角，因此需要不断进行方向调整，而 DRL-CPVO-CBFs 仅需要考虑机器人跟踪范围内的关键障碍物，所以路径更加平滑；同时没有加入 VO 奖励项的 DRL 算法路径也是较平滑的，因为不需要主动寻找最优的航向角度，但是该算法生成的路径较长。此外，由线速度和角速度分布直方图可以进一步看出所提算法对机器人执行的线速度进行了安全限制，最大速度比例略微下降，同时角速度分布更加均匀，从而速度更加稳定。虽然导航过程中部分动作速度变小，但是能够获得更加安全的运动和更短的路径。因此，DRL-CPVO-CBFs 实现了复杂多动态障碍物环境下的安全高效导航。

3.3　真实环境测试

真实环境测试所用实验平台如图 11 所示，该机器人的主控使用的是 NVIDIA Orin Nano 8 GB，系统为 Ubuntu20.04，ROS 版本为 ROS1 Noetic。为了更好的部署在该主控上，使用 TensorRT8.5.2，CUDA 版本为 11.4。激光雷达使用的是思岚科技 RPLIDAR C1，两驱动轮电机是由 STM32 控制的。

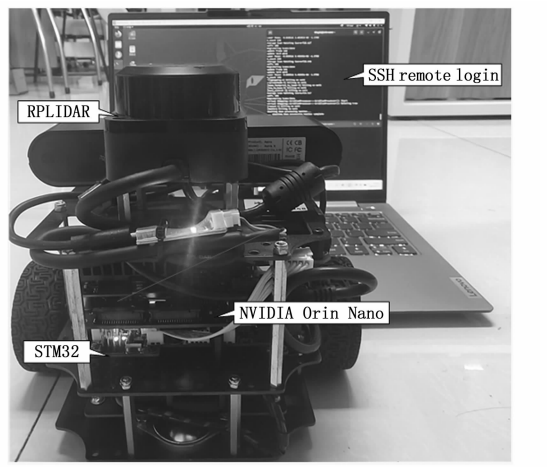

图 11　移动机器人实验平台

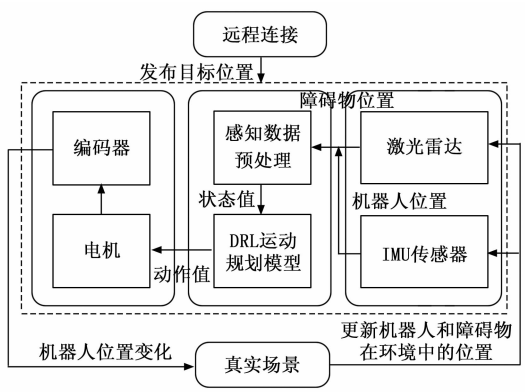

图 12　机器人实验部署框架图

DRL 导航实验部署框架如图 12 所示，首先通过 PC 端远程连接移动机器人，用来发送机器人所要到达的目标位置，同时运行驱动电机、编码器等底层硬件模块和激光雷达、IMU 传感器等感知模块。然后运行感知数据预处理和加载训练好的导航模型两部分代码，感知数据预处理是将传感器获取的信息按设计的状态空间进行计算得到模型的状态输入，模型接收状态值并生成动作值。将动作值包含的线速度和角速度转换为驱动电机的转速，从而控制机器人的速度大小和方向，同时编码器返回实际位置和速度信息。机器人在真实环境中的位置和速度不断变化，感知模块不断获取新的状态信息，并进行下一步的计算处理，如此一步步循环完成机器人在复杂多动态障碍物环境中的导航。

DROW 是一种基于深度学习的二维激光雷达扫描行人检测算法，广泛用于各种室内外服务型机器人，输出为一维的行人标签和对应的二维坐标。通过 DROW 算法检测到行人目标后，再应用所提导航策略于行人真

实场景中。为了验证算法在真实行人环境下的有效性，首先远程控制两轮差速移动机器人使用激光雷达构建了实验室走廊环境的二维地图并保存。然后让机器人在走廊中依次导航至所设定的 5 个不同目标点序列，目标点分布尽量按 S 型曲线分布，这样既能保证不同目标之间的距离，又能使机器人面对不同的环境状态，防止因为行人主动让路干扰实验结果。

使用 DRL-CPVO-CBFs 算法进行实验，结果显示机器人以 0.2 m/s 的平均速度移动了 6.39 m，成功到达了所有目标点。因此，在这种动态变化的走廊环境中，机器人可以针对关键移动行人找到最优的运动方向，并把与行人间的距离限制在一定范围内，安全、快速完成给定导航任务。在行人较少情况下，机器人表现比较轻松，速度更快；在行人较多时，机器人表现的更加稳定，在保证安全性的前提下最大化导航效率，虽然速度略低但是轨迹也更短，不会远离最短路径，导航过程中不同时刻示意图如图 13 所示。

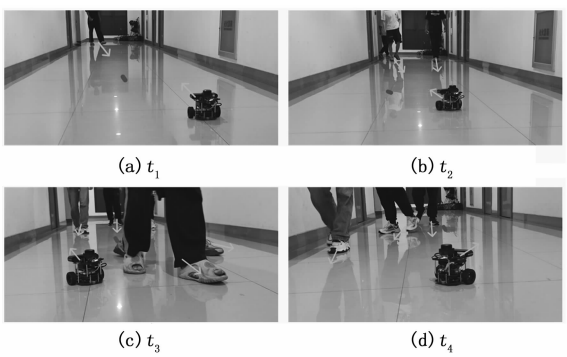

图 13 实验室走廊环境下机器人在不同时刻的反应

4 结束语

本文针对移动机器人在复杂多动态障碍物环境中导航效率低且缺乏安全保障的问题，提出了一种带有安全屏障的深度强化学习导航算法。利用风险评估的思想，设计导航策略的状态空间和奖励函数，降低感知和决策过程中的时间成本。同时为保证策略在该环境下应用的安全性，利用 CBF 函数约束机器人的安全距离。通过搭建的仿真环境测试以及真实行人环境测试，结果表明所提算法在成功率、时间成本和安全得分等指标都有明显提升，实现了移动机器人复杂多动态障碍物环境下的安全高效导航。

参考文献：

[1] 周 济. 智能制造——“中国制造 2025”的主攻方向 [J]. 中国机械工程，2015，26 (17)：2273 - 2284.

[2] RUBIO F，VALERO F，LLOPIS-ALBERT C. A review of mobile robots：concepts，methods，theoretical framework，and applications [J]. International Journal of Advanced Robotic Systems，2019，16 (2)：1 - 22.

[3] SPRUNK C，LAU B，PFAFF P，et al. An accurate and efficient navigation system for omnidirectional robots in industrial environments [J]. Autonomous Robots，2017，41 (2)：473 - 493.

[4] 翟敬梅，刘 坤，徐 晓. 室内移动机器人自主导航系统设计与方法 [J]. 计算机集成制造系统，2020，26 (4)：890 - 899.

[5] 梅 柯，锁少伟，王 静，等. 麦克娜姆轮 AGV 在汽车焊装车身转运中的应用 [J]. 自动化应用，2018 (5)：10 - 13.

[6] YANG H，DING L，GAO H，et al. High-fidelity dynamic modeling and simulation of planetary rovers using single-input-multi-output joints with terrain property mapping [J]. IEEE Transactions on Robotics，2022，38 (5)：3238 - 3258.

[7] MARTINI M，PÉREZ-HIGUERAS N，Ostuni A，et al. Adaptive social force window planner with reinforcement learning [C] // 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Abu Dhabi，UAE：IEEE，2024：4816 - 4822.

[8] LI Z，JING X，SUN B，et al. Autonomous navigation of a tracked mobile robot with novel passive bio-inspired suspension [J]. IEEE/ASME Transactions on Mechatronics，2020，25 (6)：2633 - 2644.

[9] NIROUI F，ZHANG K，KASHINO Z，et al. Deep reinforcement learning robot for search and rescue applications：exploration in unknown cluttered environments [J]. IEEE Robotics and Automation Letters，2019，4 (2)：610 - 617.

[10] 孙玉玺. 轮式移动机器人自主探索与运动规划方法研究 [D]. 济南：山东大学，2023.

[11] ZHU Z，HU C，ZHU C，et al. An improved dueling deep double-q network based on prioritized experience replay for path planning of unmanned surface vehicles [J]. Journal of Marine Science and Engineering，2021，9 (11)：1267.

[12] GUO S，ZHANG X，DU Y，et al. Path planning of coastal ships based on optimized DQN reward function [J]. Journal of Marine Science and Engineering，2021，9 (2)：210.

[13] WEN S，LV X，LAM H K，et al. Probability dueling DQN active visual SLAM for autonomous navigation in indoor environment [J]. Industrial Robot：The International Journal of Robotics Research and Application，2021，48 (3)：359 - 365.

[14] HESSEL M，MODAYIL J，VAN HASSELT H，et al. Rainbow：combining improvements in deep reinforcement learning [C] // AAAI conference on artificial intelligence，2018.

[15] HUANG C, CHEN G, GONG Y, et al. Buffer-aided relay selection for cooperative hybrid NOMA/OMA networks with asynchronous deep reinforcement learning [J]. IEEE Journal on Selected Areas in Communications, 2021, 39 (8): 2514 - 2525.

[16] JING A, TANG Z, GAO J, et al. An improved DDPG reinforcement learning control of underwater gliders for energy optimization [C] // Harbin, China, 2020.

[17] CHEN Y, LIANG L. SLP-improved DDPG path-planning algorithm for mobile robot in large-scale dynamic environment [J]. Sensors, 2023, 23 (7): 3521.

[18] ZHANG S, TANG W, LI P, et al. Mapless path planning for mobile robot based on improved deep deterministic policy gradient algorithm [J]. Sensors, 2024, 24 (17): 5667.

[19] TAI L, PAOLO G, LIU M. Virtual-to-real deep reinforcement learning: Continuous control of mobile robots for mapless navigation [C] // Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2017.

[20] GAO J , YE W , GUO J, et al. Deep reinforcement learning for indoor mobile robot path planning [J]. Sensors, 2020, 20 (19): 5493.

[21] LIU H, SHEN Y, ZHOU C, et al. TD3 based collision free motion planning for robot navigation [C] // Proceedings of the 2024 6th International Conference on Communications, Information System and Computer Engineering (CISCE). IEEE, 2024: 247 - 250.

[22] TAN Y, LIN Y, LIU T, et al. PL-TD3: a dynamic path planning algorithm of mobile robot [C] //Proceedings of the 2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC). IEEE, 2022: 3040 - 3045.

[23] ZHANG X, ZHANG H, ZHOU H T, et al. Safe reinforcement learning with dead-ends avoidance and recovery [J]. IEEE Robotics and Automation Letters, 2023, 9 (1): 491 - 498.

(上接第 298 页)

[9] YANG Y, PRÖBSTING S, LIU Y, et al. Effect of dual vortex shedding on airfoil tonal noise generation [J]. Physics of Fluids, 2021, 33 (7): 075102.

[10] YU L, YU L, WANG R, et al. Passive sound detection of the helicopter in the far-field with a spectral coherence decomposition method [J]. Mechanical Systems and Signal Processing, 2023, 185: 109754.

[11] ZHAO K, ZHANG R, KOPIEV V, et al. A parametric study of nose landing gear noise in a large-scale aeroacoustic wind tunnel [J]. Applied Acoustics, 2022, 201: 109130.

[12] ZAYTSEV M YU, KOPIEV V F, VELICHKO S A, et al. Localization and ranking of aircraft noise sources in flight tests and comparison with acoustic measurements of a large-scale wing model [J]. Acoustical Physics, 2023, 69 (2): 182 - 192.

[13] YU L, WU H, ANTONI J, et al. Extraction and imaging of aerodynamically generated sound field of rotor blades in the wind tunnel test [J]. Mechanical Systems and Signal Processing, 2019, 116: 1017 - 1028.

[14] YU L, GONG Z, CHU N, et al. Adaptive imaging of sound source based on total variation prior and a subspace iteration integrated variational bayesian method [J]. IEEE Transactions on Instrumentation and Measurement, 2021, 70: 1 - 17.

[15] GUO Y, ZHOU Y, GUAN L, et al. Acoustic helicopter recognition via convolutional neural network [J]. 2018 IEEE 23rd International Conference on Digital Signal Processing (DSP), 2018, 00: 1 - 5.

[16] CHU N, NING Y, YU L, et al. Acoustic source localization in a reverberant environment based on sound field morphological component analysis and alternating direction method of multipliers [J]. IEEE Transactions on Instrumentation and Measurement, 2021, 70: 1 - 13.

[17] LEE J H, SEO J S. Application of spectral kurtosis to the detection of tip vortex cavitation noise in marine propeller [J]. Mechanical Systems and Signal Processing, 2013, 40 (1): 222 - 236.

[18] ANTONI J. Fast computation of the kurtogram for the detection of transient faults [J]. Mechanical Systems and Signal Processing, 2007, 21 (1): 108 - 124.

[19] RAMOS-ROMERO C, GREEN N, TORIJA A J, et al. On-field noise measurements and acoustic characterisation of multi-rotor small unmanned aerial systems [J]. Aerospace Science and Technology, 2023, 141: 108537.

[20] WANG Q, ZHENG F, QIAN W, et al. A practical filter error method for aerodynamic parameter estimation of aircraft in turbulence [J]. Chinese Journal of Aeronautics, 2023, 36 (2): 17 - 28.

[21] ELSHAFEI M, AKHTAR S, AHMED M S. Parametric models for helicopter identification using ANN [J]. IEEE Transactions on Aerospace and Electronic Systems, 2000, 36 (4): 1242 - 1252.

[22] YU D, LI J. Recent progresses in deep learning based acoustic models [J]. IEEE/CAA Journal of Automatica Sinica, 2017, 4 (3): 396 - 409.

[23] SLIMENE M B, OUALI M-S. Anomaly detection method of aircraft system using multivariate time series clustering and classification techniques [J]. IFAC-Papers on Line, 2022, 55 (10): 1582 - 1587.