第54卷第1期 南京航空航天大学学报 Vol. 54 No. 1
2022年2月 Journal of Nanjing University of Aeronautics & Astronautics Feb. 2022

DOI:10.16356/j.1005-2615.2022.01.005

基于DDPG-PID的机器人动态目标跟踪与避障控制研究

刘 勇,李 祥,蒋沛阳,孙博熙,吴 喆,姜 潇,钱 森
(合肥工业大学机械工程学院,合肥 230009)

摘要: 针对机器人在动态复杂环境下的操作问题,研究机械臂跟踪运动目标及避障的运动控制方法。采用传统PID控制方法与DDPG强化学习算法相结合的方式,利用PID控制使得机械臂的工作平面快速接近目标物并与之重合,再使用DDPG算法让机械臂在平面内自主学习追踪目标物投影同时避开障碍物投影,最终在三维空间中实现跟踪与避障。该方法将机械臂作为一个智能体,智能体通过感知目标物和障碍物的距离偏差来自主学习避障跟踪策略。将本方法用于多自由度机械臂路径规划与避障任务中,对比单纯使用DDPG算法将机械臂作为智能体在空间中进行决策控制,仿真结果显示本文所提出的方法收敛效果和控制性能更好,机械臂能在训练后实现目标物的稳定跟踪和障碍物的躲避。

关键词: 强化学习;PID控制;路径规划;避障
中图分类号: TP242 **文献标志码:** A **文章编号:** 1005-2615(2022)01-0041-10

Research on Robot Dynamic Target Tracking and Obstacle Avoidance Control Based on DDPG-PID

LIU Yong, LI Xiang, JIANG Peiyang, SUN Boxi, WU Zhe, JIANG Xiao, QIAN Sen
(School of Mechanical Engineering, Hefei University of Technology, Hefei 230009, China)

Abstract: Aiming at the operational problems of robots in dynamic and complex environments, the motion control method of manipulator tracking moving targets and avoiding obstacles is studied. The traditional PID control method is combined with DDPG algorithm. PID control is used to make the working plane of the manipulator approach the target quickly and coincide with it. Then DDPG algorithm is used to make the manipulator autonomously learn to track the projection of target and avoid the projection of obstacles in the plane, and finally achieve tracking and obstacle avoidance in the three-dimensional space. This method takes the manipulator as an agent which perceives the distance deviation between the target and the obstacle to learn obstacle avoidance and tracking strategies automatically. This method is applied to the path planning and obstacle avoidance task of multi-degree-of-freedom manipulator. Compared with DDPG algorithm, which only takes manipulator as agent to make decision control in space, the simulation results show that the proposed method has a better convergence effect and control performance, and the manipulator can stably track the target and avoid obstacles after training.

Key words: reinforcement learning; PID control; path planning; obstacle avoidance

如何使机器人在完成动态环境下复杂操作任务的同时保证机器人本体不受损害是机器人工程

基金项目: 国家自然科学基金(52175013);中央高校基本科研项目(JZ2020HGTB0034)。
收稿日期: 2021-12-30;**修订日期:** 2022-01-15
通信作者: 钱森,男,博士,副教授,E-mail: qiansenhfut@126.com。

引用格式: 刘勇,李祥,蒋沛阳,等. 基于DDPG-PID的机器人动态目标跟踪与避障控制研究[J]. 南京航空航天大学学报,2022,54(1):41-50. Research on robot dynamic target tracking and obstacle avoidance control based on DDPG-PID[J]. Journal of Nanjing University of Aeronautics & Astronautics,2022, 54(1):41-50.

应用中的核心问题。针对这个问题,众多学者展开了如何控制机械臂在复杂工作环境下从初始点运动到目标点并能避开所有障碍物的研究,即避障轨迹规划研究[1]。传统机器人控制方法具有稳定可靠的特点而获得了广泛的应用。祝敬等[2]采用快速搜索随机树方法对人工势场法进行局部改进以解决人工势场法易陷入局部最小点的问题,最后快速有效地规划出一条无障碍的路径。马宇豪等[3]提出了一种基于关节空间的6次多项式轨迹规划的机械臂避障算法,在机械臂运动学、轨迹长度、关节转动角度等约束下采用多目标优化遗传算法对参数优化获得最优轨迹。Wang等[4]设计了一种用于空间漂浮7自由度冗余机械臂的具有避障功能的非线性模型预测控制策略,该策略将避障问题转化为线性不等式约束,并将其集成到二次规划优化问题中。面向动态复杂的操作环境,机器人呈现出工作空间到关节空间映射模型复杂和状态信息误差大的特点,仅使用传统的反馈控制方法进行避障对运动学与动力学及环境模型依赖程度较高,在面对未知可变环境的快速响应方面性能不足。

近年来,随着人工智能的兴起和机器学习的日益发展[5],衍生出基于强化学习[6]、示教学习和小数据学习的机器人操作方法[7]。将人工智能和机器人技术相结合,使用强化学习为机器人提供框架和工具,从而解决机器人的复杂控制问题[8]。强化学习算法通过利用在马尔可夫决策过程中寻找到的最优策略建立机器人状态与动作间的映射关系,与控制系统决策进行互补以面对复杂多变的操作环境[7],并且在机器人控制领域取得了一定的成果。李鹤宇等[9]利用DDPG(深度确定性梯度策略)算法控制虚拟环境下的机械臂,实现了相比人工调试更快的抓取动作。Sarantopoulos等[10]提出了一种具有更快收敛性的模块化的DQN(深度Q网络)算法,并结合DDPG算法用于实现机械臂将目标从杂物中分离,为抓取目标创造空间。徐帷等[11]针对6自由度空间机械臂在线路径规划问题,利用Sarsa(λ)强化学习算法实现了目标跟踪及避障的自主路径规划,但在解决具有冗余自由度机械臂的连续运动时可能存在局限性和繁琐性。Christen等[12]提出了一种具有泛化能力的层次强化学习框架——HiDe,将机器人的复杂控制分解为全局规划和低级控制,DQN和DDPG算法分别用于训练规划层与控制层,在仿真中成功控制机械臂推着小球避开障碍物到达目的地。Sangiovanni等[13]通过归一化优势函数,利用基于神经网络的Q-learning强化学习算法控制机械臂,在一个存在不可预知障碍物的连续空间中避开障碍物到达指定位置。

上述基于强化学习的方法虽然能够使机器人具备自主规划能力,但在一般情况下,由于机器人动作空间维数高和状态信息误差大的原因,训练过程可能陷入局部最优或出现学习能力不足的问题。针对这一困境,需恰当地融入先验知识以降低搜索空间维数或提升数据利用率[8],通过尽可能少的尝试实现机器人强化学习决策。Chatzilygeroudis等[14]提出了微数据强化学习,该算法可通过在策略结构与参数,期望奖励模型和状态转移动力学模型3个方面融入先验知识或者构建替代模型实现机器人在数据绝对少的情况下学习到相关策略。Zhong等[15]设计了一种基于DDPG算法和机械臂逆运动学的混合算法,用以解决机器人的避障路径规划问题,该算法将机械臂的逆运动学作为一种先验知识融入DDPG算法,使得其收敛性和收敛速度大大提高。Lin等[16]考虑到机器人末端轨迹在执行任务中的对称性,结合KER(Kaleidoscope experience replay)和GER(Goal-augmented experience replay)两种经验回放机制的优势与不足,提出了一个用于在DDPG算法中实现数据增强的框架——ITER(Invariant transform experience replay),实验表明带有数据增强的DDPG算法可以大大提高机械臂在有障碍和无障碍的情况下学习推、滑和取放物体的效率。

强化学习通过融入先验知识或提高数据质量使得策略网络的性能有所提升,然而这种提升十分有限。众多学者注意到将强化学习算法结合传统控制方法与控制理论能进一步提升算法性能,改善控制效果。一方面,传统控制方法和强化学习算法可以根据机械臂和环境的状态分别对机械臂进行控制,以提高整体控制效果。Johannink等[17]研究了一种将真实机械臂的复杂控制问题(涉及与环境接触)分解成能用常规反馈控制方法(阻抗控制)有效求解的部分,以及可以用不同强化学习算法求解难以求解的剩余部分,最后叠加两个控制信号得到最终的控制策略。Yamada等[18]通过基于采样的运动规划器增强的无模型强化学习算法,使得机械臂能够根据动作大小在直接输出单步动作和调用运动规划器之间平滑转换从而提高了机械臂避障轨迹规划的效率和安全性。另一方面,传统控制方法和控制理论可以作为一种辅助手段来从不同方面对强化学习算法进行优化,提高其性能和效率。Al-gabalawy等[19]结合模型预测控制和TRPO(置信域梯度策略优化)算法,构建了一种混合算法以

实现机械臂在有环境边界约束和障碍物约束的情况下到达目标位置，该混合算法相比于无模型强化学习算法所需的样本数据更少，性能更高。Kukker等[20]提出了一种基于随机遗传算法辅助的模糊Q-learning的机械臂控制方法，遗传算法作为随机优化器，对基于模糊Q-learning的控制器各阶段的动作选择进行优化。Ota等[21]提出了一种将TD3（双延迟深度确定性策略梯度）算法与传统快速搜索随机树算法相结合的方法，训练机械臂获得动态平滑且能够避开障碍的最优轨迹，其中快速搜索随机树算法用于指导智能体学习，即限制搜索域以提高算法收敛性和收敛速度。

本文采用DDPG强化学习算法与PID控制相结合的方式求解机械臂动态目标跟踪与自主避障的运动策略。一方面，为提升DDPG强化学习算法的收敛性，本文结合机械臂的机构特性，将目标物与障碍物投影到机械臂的工作平面构建虚拟目标物与虚拟障碍物来降低强化学习动作空间的维数；另一方面，引入传统PID控制方法来控制机械臂的工作平面快速接近目标，提高目标跟踪与避障的成功率。

1　问题描述和机械臂几何关系分析

1.1　运动路径规划问题描述

如图1所示的机械臂是一种经典的6自由度机械臂——UR5机械臂，其结构包括底座、两个肩关节和一个位于中间的肘关节以及末端3个腕关节。

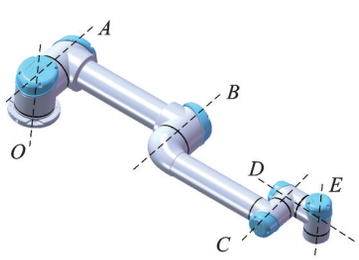


图1　UR5机械臂模型

Fig.1　Model of UR5 manipulator

UR5机械臂的运动由6个关节所共同控制，每个关节控制的主要目标有所不同。在机械臂的末端加装执行器后，末端关节D、E主要决定了末端执行器在完成特定任务时的姿态，在进行大幅度运动或完成常规的目标抓取等信息捕获任务时，主要依赖于O、A、B、C关节进行快速运动，使末端执行器快速接近目标。本文主要考虑机器人大范围运动目标跟踪的粗捕获任务[11]，将关节D、E固定且在建模过程中引入关节偏置从而更加接近真实机械臂模型，图1简化后的空间数学模型如图2所示。

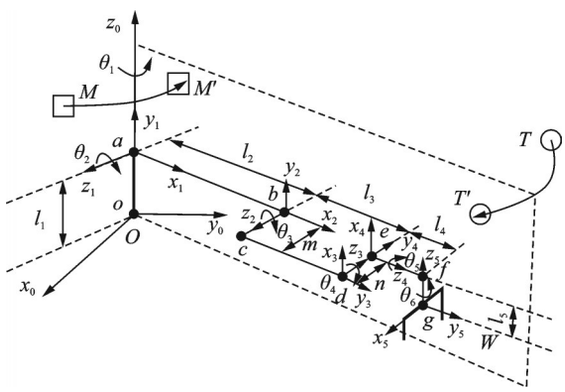


图2　机械臂模型简化

Fig.2　Simplification of manipulator model

$Ox_0y_0z_0$是基准坐标系，机械臂的基座底面中心与该坐标系原点O重合，W为经过z_0轴与ab杆平行且与x_0Oy_0平面垂直的机械臂工作平面，由肩关节o的关节角θ_1确定。各连杆长度为l_1、l_2、l_3、l_4和l_5。本文仅考虑两个肩关节o和a、一个肘关节b、一个腕关节d，它们相对于其零位的关节角为θ_1、θ_2、θ_3和θ_4。T为目标物，M为障碍物，它们绕z_0轴旋转在平面W内的旋转投影构成虚拟目标物T'和虚拟障碍物M'。

机械臂避障路径规划的目标是实现机械臂末端点g从初始位置到达目标物T的位置，同时在整个过程中不与环境中的障碍物M发生碰撞。为实现这一目标，可以将三维空间跟踪避障问题分解：一方面，控制机械臂的肩关节o，使得工作平面可以快速到达目标物所在的平面，同时，目标物T与障碍物M的投影T'与M'一直作为虚拟目标物与虚拟障碍物存在于工作平面W中；另一方面，在平面W中，通过控制关节a、b和d使得机械臂末端点g能够跟踪虚拟目标物T'同时避开虚拟障碍物M'。通过上述两方面的协同控制，最终在三维空间中实现机械臂末端点g跟踪目标物T，同时避开障碍物M。

1.2　平面几何关系分析

如图3所示，将整个机械臂先投影到工作平面W，在该平面内重新建立坐标系xOy计算机械臂各关节与末端的位置关系。

关节a、b、d、f及机械臂末端点g在该平面坐标系中的位置关系可表示为

$$\begin{bmatrix} x_a \\ y_a \end{bmatrix} = \begin{bmatrix} 0 \\ l_1 \end{bmatrix} \tag{1}$$

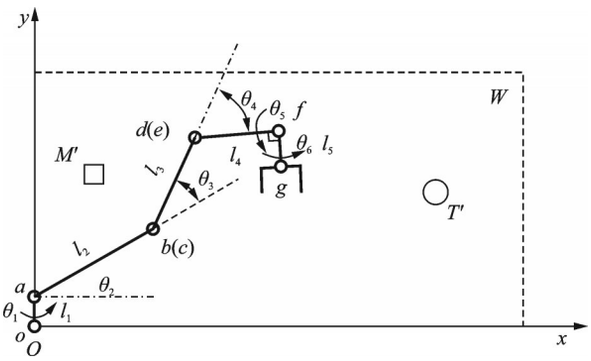


图3 机械臂在工作平面的投影

Fig.3 Projection of manipulator on the working plane

$$\begin{bmatrix} x_b \\ y_b \end{bmatrix}=\begin{bmatrix} l_2\cos\theta_2 \\ l_1+l_2\sin\theta_2 \end{bmatrix} \tag{2}$$

$$\begin{bmatrix} x_d \\ y_d \end{bmatrix}=\begin{bmatrix} x_b+l_3\cos(\theta_2+\theta_3) \\ y_b+l_3\sin(\theta_2+\theta_3) \end{bmatrix} \tag{3}$$

$$\begin{bmatrix} x_f \\ y_f \end{bmatrix}=\begin{bmatrix} x_d+l_4\cos(\theta_2+\theta_3+\theta_4) \\ y_d+l_4\sin(\theta_2+\theta_3+\theta_4) \end{bmatrix} \tag{4}$$

$$\begin{bmatrix} x_g \\ y_g \end{bmatrix}=\begin{bmatrix} x_f+l_5\sin(\theta_2+\theta_3+\theta_4) \\ y_f-l_5\cos(\theta_2+\theta_3+\theta_4) \end{bmatrix} \tag{5}$$

机械臂的工作平面 W 为一个经过 z_0 轴与 ab 杆平行且与 x_0Oy_0 平面垂直的一个平面，此时机械臂末端点 g 不在工作平面上，即发生错位从而导致末端点 g 不能准确与目标物的坐标重合，这是机械臂关节自身宽度导致的。为解决这一问题，本文对关节 o 的关节角 θ_1 做了偏置处理，即将工作平面 W 向着 g 点绕 z_0 轴旋转 α 度得到一个新的工作平面 W'，使得末端点 g 在工作平面上。选用工作平面 W'，一方面可以使得PID对关节 o 的控制（即对工作平面的控制）更加直接，简化后续目标物的投影转换；另一方面可以平衡工作平面两侧UR5机械臂的构型分布以简化对避障方法的设计。偏置只会影响肩关节 o 的角度变化，不影响强化学习算法对 a、b 和 d 关节的控制。针对UR5机械臂构型特点，将机械臂向 x_0Oy_0 平面进行投影，建立如图4所示的关节偏置模型。与图2相比，该模型参考了机械臂的实际模型，关节 o 与关节 a 在 x_0 轴方向上存在一个大小为 p 的偏差。

偏置距离为

$$d_\Delta=p-m+n \tag{6}$$

$$s=l_2\cos\theta_2+l_3\cos(\theta_2+\theta_3)+l_4\cos(\theta_2+\theta_3+\theta_4)+l_5\sin(\theta_2+\theta_3+\theta_4) \tag{7}$$

解得偏置角为为

$$\alpha=\arctan\frac{d_\Delta}{s} \tag{8}$$

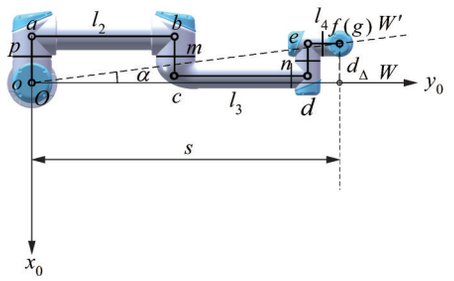


图4 关节偏置

Fig.4 Joint offset

2 强化学习及其状态与奖励函数设计

本文应用DDPG强化学习算法，将机械臂视为能感知环境状态并输出动作决策的agent，agent的动作 $\boldsymbol{a}$ 为相应的关节角速度；环境状态 $\boldsymbol{s}$ 为要实现目标所需要反馈给agent的各种相关的环境信息；奖励函数 R 为机械臂agent在与环境交互过程中所获得的回报，奖励函数的设计非常重要，能直接影响后面学习过程的收敛性。

2.1 DDPG强化学习算法

DDPG是一种基于actor-critic（演员-评论家）框架的强化学习算法，它可以应对agent需要输出连续动作的问题，更加符合机械臂在状态空间内运动连续的特性，因此本文采用该算法进行求解。图5描述了DDPG算法的网络框架，s_t、r_t、a_t 为 t 时间步的状态、奖励以及动作；μ 表示策略，θ 为神经网络参数。

评论家（critic）网络用于拟合 Q 函数（价值函数），包含有两个 Q 神经网络拷贝，online和target，记为 $Q(s,a|\theta^Q)$ 和 $Q'(s,a|\theta^{Q'})$，两者的参数 θ^Q 和 $\theta^{Q'}$ 的初始值相同。在训练过程中，当记忆库中的样本数到达设定数值时，则从中提取数量为 N 的样本：$N\times(s_i,a_i,r_i,s_{i+1})$，并通过最小化式（9）所示的损失函数 L 来更新online Q 网络的参数 θ^Q。

$$L=\frac{1}{N}\sum_i\left(y_i-Q\left(s_i,a_i|\theta^Q\right)\right)^2 \tag{9}$$

$$y_i=r_i+\gamma Q'\left(s_{i+1},\mu'\left(s_{i+1}|\theta^{\mu'}\right)|\theta^{Q'}\right) \tag{10}$$

演员（actor）网络用于拟合确定性策略函数，该函数通过当前的状态选择合理的动作作为输出。对比评论家（critic），演员（actor）也有一个online神经网络和一个target神经网络，两个网络的参数 θ^μ 和 $\theta^{\mu'}$ 的初始值相同。通过计算式（11）所示的策略梯度来更新online策略网络的参数 θ^μ。

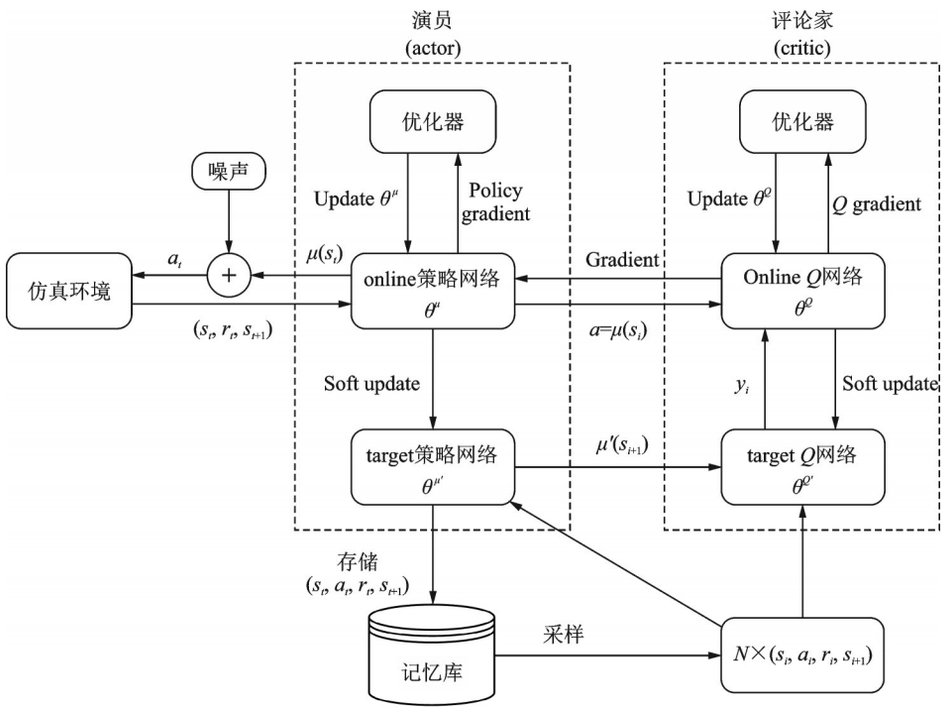


图5　DDPG算法网络框架

Fig.5　Network framework of DDPG algorithm

$$\nabla_{\theta^{\mu}} J \approx E_{s_t\sim\rho^{\beta}}\left[\nabla_{\theta^{\mu}} Q(s,a|\theta^{Q})\mid_{s=s_t,a=\mu(s_t|\theta^{\mu})}\right]=$$

$$E_{s_t\sim\rho^{\beta}}\left[\nabla_a Q(s,a|\theta^{Q})\mid_{s=s_t,a=\mu(s_t)}\nabla_{\theta_{\mu}}\mu(s|\theta^{\mu})\mid_{s=s_t}\right] \quad (11)$$

应用Adam(Adaptive moment estimation)算法取代随机梯度下降法来更新θ^{Q}和θ^{μ}，最后通过soft update算法更新target Q网络的参数$\theta^{Q'}$和target策略网络的参数$\theta^{\mu'}$。

$$\text{soft update:}\begin{cases}\theta^{Q'}\leftarrow\tau\theta^{Q}+(1-\tau)\theta^{Q'}\\ \theta^{\mu'}\leftarrow\tau\theta^{\mu}+(1-\tau)\theta^{\mu'}\end{cases} \quad (12)$$

式中τ一般取0.001。

2.2　环境状态设计

本文中强化学习agent的动作空间维数为3，每一个动作包含3个关节角的角速度$\dot{\theta}_2$、$\dot{\theta}_3$与$\dot{\theta}_4$，即$\boldsymbol{a}=[\dot{\theta}_2,\dot{\theta}_3,\dot{\theta}_4]$。环境的状态需要实时反馈给agent，以便agent根据这一反馈信息做出动作。结合1.1和1.2节的分析，环境状态设计如下：

(1) 关节o、关节a、关节b和关节d相对于其零位的关节角分别为θ_1、θ_2、θ_3与θ_4；

(2) 机械臂末端点g，虚拟目标物T'及虚拟障碍物M'在工作平面的位置分别为(x_g,y_g)、$(x_{T'},y_{T'})$和$(x_{M'},y_{M'})$；

(3) 用于判断机械臂末端点g是否到达目标点范围的一个布尔变量为g_t，若到达目标点范围，$g_t=1$，否则$g_t=0$。

式(5)已经给出机械臂末端点g的位置(x_g,y_g)，$(x_{T'},y_{T'})$与$(x_{M'},y_{M'})$分别为虚拟目标物T'与虚拟障碍物M'的位置。本文所设定的目标点在虚拟目标物T'上边界垂直向上10 mm处(本文公式中所涉及的距离单位均为mm)。该目标点记为ξ，其在工作平面中的位置为(x_ξ,y_ξ)，其中$x_\xi=x_{T'}$，$y_\xi=y_{T'}+10$。利用机械臂末端点g与目标点ξ的距离$d_{g\xi}$来确定g_t的值，$d_{g\xi}$的计算如式(13)所示。

$$d_{g\xi}=\sqrt{(x_g-x_\xi)^2+(y_g-y_\xi)^2} \quad (13)$$

$$\begin{cases}g_t=1 & d_{g\xi}<20\text{ mm}\\ g_t=0 & \text{其他}\end{cases} \quad (14)$$

2.3　碰撞检测与奖励函数设计

避障过程要求整个机械臂不能与障碍物发生碰撞，通过计算虚拟障碍物M'与机械臂在工作平面内的绝对距离来检测碰撞。

如图6所示，当$\alpha_{aM'}\leqslant 90^{\circ}$且$\alpha_{bM'}\leqslant 90^{\circ}$时，直接计算虚拟障碍物$M'$与连杆$ab$的最小距离$d_{abM'}$；当$\alpha_{aM'}>90^{\circ}$时，$d_{abM'}=|aM'|$；当$\alpha_{bM'}>90^{\circ}$时，$d_{abM'}=|bM'|$。$d_{abM'}$按式(15)计算，$\alpha_{aM'}$与$\alpha_{bM'}$按式(16)计算。

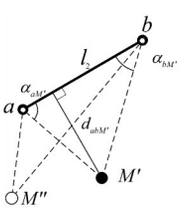


图6　碰撞检测

Fig.6　Collision detection

$$d_{abM'}=\begin{cases}\sqrt{(x_a-x_{M'})^2+(y_a-y_{M'})^2}\sin\alpha_{aM'} & \alpha_{aM'}\leqslant 90^\circ\text{且}\alpha_{bM'}\leqslant 90^\circ\\ \sqrt{(x_a-x_{M'})^2+(y_a-y_{M'})^2} & \alpha_{aM'}>90^\circ\\ \sqrt{(x_b-x_{M'})^2+(y_b-y_{M'})^2} & \alpha_{bM'}>90^\circ\end{cases}\tag{15}$$

$$\begin{cases}\alpha_{aM'}=\arccos\dfrac{[(x_a-x_{M'})^2+(y_a-y_{M'})^2]+l_2^2-[(x_b-x_{M'})^2+(y_b-y_{M'})^2]}{2l_2\sqrt{(x_a-x_{M'})^2+(y_a-y_{M'})^2}}\\ \alpha_{bM'}=\arccos\dfrac{[(x_b-x_{M'})^2+(y_b-y_{M'})^2]+l_2^2-[(x_a-x_{M'})^2+(y_a-y_{M'})^2]}{2l_2\sqrt{(x_b-x_{M'})^2+(y_b-y_{M'})^2}}\end{cases}\tag{16}$$

连杆cd、ef与fg的碰撞检测亦如此处理，可分别得到$d_{cdM'}$、$d_{efM'}$和$d_{fgM'}$。在真实环境中，传统控制方法通过反复调试可以保证机械臂末端在跟踪目标物的同时不与目标物发生碰撞，而强化学习要求机械臂自己探索出目标跟踪路径，这可能使得机械臂在跟踪目标物的同时会与目标物发生碰撞。因此，本文添加了机械臂对虚拟目标物T'的碰撞检测。碰撞检测方法与上述方法相同，连杆ab、cd、ef与fg与目标物的距离分别为$d_{abT'}$、$d_{cdT'}$、$d_{efT'}$和$d_{fgT'}$。最后，将$\min\{d_{abM'},d_{cdM'},d_{efM'},d_{fgM'}\}$和$\min\{d_{abT'},d_{cdT'},d_{efT'},d_{fgT'}\}$引入强化学习的奖励函数中，利用强化学习算法实现避障。

在强化学习中，奖励来自于环境，面对不同的任务，奖励函数需要根据任务特性和环境状态仔细设计[6]。仅考虑每回合机械臂末端点是否到达目标点范围所设计的奖励函数是一种稀疏奖励，但这样的奖励函数会使agent很难获得奖励，算法收敛性差，导致模型学习缓慢甚至无法收敛。为此，本文设计了一种如式(17)所示的连续奖励函数R

$$R=R_1+R_2+R_3+R_4+R_5\tag{17}$$

式(17)中，R_1与$d_{g\xi}$有关，用于引导机械臂末端点g到达目标点范围，按式(18)计算。R_2用于引导机械臂的连杆fg尽量保持竖直朝向目标物的状态，按式(19)计算。

$$R_1=-0.02d_{g\xi}+\frac{10}{0.03d_{g\xi}+1}\tag{18}$$

$$R_2=0.05(y_f-y_g)\tag{19}$$

R_3用于引导机械臂避开虚拟障碍物M'，按式(20)计算。

$$R_3=-10\lambda\tanh[0.05\times(80-\min\{d_{abM'},d_{cdM'},d_{efM'},d_{fgM'}\})+1]\tag{20}$$

然而，仅按照上述方式可能会产生如下误判：在三维空间中障碍物并未与机械臂发生干涉，但基于其在工作平面的投影判定当前状态会发生碰撞，由此会导致机械臂脱离原有跟踪运动轨迹进行非必要的避障运动。针对该类问题，引入了如式(21)所示的避障纠正因子λ。其中，$\theta_\Delta=|\theta_1-\arctan(y_M/x_M)|$为工作平面与障碍物所在平面的夹角。当$\theta_\Delta>\theta_b$时，$\lambda$无限接近于0使得$R_3$接近于0；而当$\theta_\Delta<\theta_b$时，$\lambda$开始激增，$R_3$被激活以引导机械臂避开障碍物。$\theta_b$为开启避障的阈值，与障碍物的尺寸、障碍物在空间中的位置及机械臂真实构型有关。R_4用于引导机械臂避免与目标物T'发生碰撞，按式(22)计算。R_5用于限制强化学习输出动作的大小，使得机械臂的运动轨迹更加平滑，按式(23)计算。

$$\lambda=0.5\tanh[5\times(0.6-\theta_\Delta)+1]\tag{21}$$

$$R_4=-5\tanh[0.05\times(40-\min\{d_{abT'},d_{cdT'},d_{efT'},d_{fgT'}\})+1]\tag{22}$$

$$R_5=-\|\boldsymbol{a}\|^2\tag{23}$$

上述奖励函数R能够根据$d_{g\xi}$、$\min\{d_{abM'},d_{cdM'},d_{efM'},d_{fgM'}\}$和$\min\{d_{abT'},d_{cdT'},d_{efT'},d_{fgT'}\}$的大小来改变变化趋势，使得模型训练更加易于收敛，获得较好的学习效果。

3 动态目标跟踪与避障控制算法流程

本文利用PID控制方法控制机械臂的第一个关节(肩关节)o。PID控制算法简单、鲁棒性好和可靠性高，经典PID控制形式离散化表示为

$$u(k)=K_{\mathrm{p}}e(k)+K_{\mathrm{i}}\sum_{n=0}^{k}e(n)+K_{\mathrm{d}}(e(k)-e(k-1))\tag{24}$$

利用式(24)，计算机械臂的工作平面(由关节o控制)与目标物的夹角作为误差e，利用误差反馈通过反复迭代尽可能减小夹角，最终使得机械臂的工作平面与目标物重合，从而将三维空间中机械臂的控制问题降维成二维平面机械臂的控制问题。

动态目标跟踪与避障控制算法流程如下。

输入环境E，状态空间S，动作空间A。

初始化actor和critic的神经网络。

从初始回合后进入循环，第1个Episode时：

(1) 初始化状态为s_0。

(2) PID控制。计算更新机械臂的工作平面和目标物的角度信息，计算角度的偏差值，通过PID控制观测上一步的误差和累计误差来给出当前步的角度偏差，使得目标物与其在工作平面的投影重合。

(3) DDPG控制。把s_t作为online策略网络的输入，计算并加入噪声N_t得到动作

$$a_t=\mu\left(s_t\right)+N_t \tag{25}$$

(4) 执行当前动作a_t得到奖励r_t和新的状态s_{t+1}，s_{t+1}作为下一步的状态值$s_t=s_{t+1}$。

(5) 将样本(s_t, a_t, r_t, s_{t+1})存储到记忆库。

(6) 记忆库达到一定的规模，从记忆库中采样N个样本$N\times(s_i, a_i, r_i, s_{i+1})$，开始学习。

(7) 利用式(10)计算当前目标价值，通过式(9)更新online Q网络参数θ^Q。

(8) 利用式(11)计算策略梯度，更新online策略网络参数θ^μ。

(9) 利用式(12)更新target Q网络和target策略网络的参数$\theta^{Q'}$和$\theta^{\mu'}$。

(10) 当达到最大步数时，则该轮训练结束，否则返回步骤2。

达到训练最大步数，第1个Episode结束，进入第2个Episode循环执行步骤1～10，直至Episode达到所设定的最大值，整个训练过程结束。

4　仿真校验

4.1　仿真参数设置

如表1所示，仿真初始参数包含UR5机械臂的初始状态、两种不同场景下目标物在空间中的初始位置、障碍物在空间中的位置。DDPG算法参数和PID算法参数分别如表2和表3所示，DDPG算法参数按照一般情况选取，PID参数经过反复调试所得。

4.2　仿真结果

本文在MATLAB环境下进行仿真测试。黄色正方体为障碍物，蓝色圆柱体为目标物。图7和图8为两种场景测试过程中机械臂状态变化。仿真测试分为两种场景，场景一为障碍物几乎不影响机械臂对目标物的跟踪；场景二为障碍物会严重影响机械臂对目标物的跟踪。

表1　仿真初始参数

Table 1　Initial simulation parameters

机械臂初始状态	目标物初始位置/mm	障碍物位置/mm
	(场景一)	
θ_1: 0°	$x_T(t)$: 500	
θ_2: −90°	$y_T(t)$: −400	
θ_3: 0°	$z_T(t)$: 200	$x_M(t)$: 400
θ_4: −90°	(场景二)	$y_M(t)$: 0
θ_5: −90°	$x_T(t)$: 250	$z_M(t)$: 200
θ_6: 0°	$y_T(t)$: −600	
	$z_T(t)$: 100	

表2　DDPG算法参数

Table 2　Parameters of DDPG algorithm

演员学习率α^a	评论家学习率α^c	折扣系数γ	τ(soft replacement)	记忆库
0.001	0.001	0.99	0.01	200 000

表3　PID参数

Table 3　Parameters of PID

比例系数K_p	积分系数K_i	微分系数K_d
10	0.01	0.01

测试场景一机械臂状态变化如图7所示，机械臂从初始位置出发，能迅速捕捉到目标物的位置并能够保持持续跟踪目标物。机械臂在跟踪目标物时会避免自身与障碍物发生碰撞。

测试场景二机械臂状态变化如图8所示，此时障碍物对机械臂跟踪目标物的运动产生严重阻碍，机械臂会优先选择满足绕开障碍物的路径而远离了目标物，当障碍物对其轨迹不产生影响时会快速跟上目标物，这满足实际中避障优先的安全性原则。

为验证避障纠正因子λ的有效性，绘制了测试场景二机械臂典型状态在工作平面投影如图9(a)所示，对应三维空间中机械臂的状态如图9(b)所示。虽然工作平面内的障碍物投影与机械臂发生干涉，但由于纠正因子的作用未产生碰撞误判，实际三维空间中机械臂并没有产生无意义的避障运动，而是保持了原有的跟踪运动轨迹。

此外，为验证所提DDPG-PID控制方法的性能，直接将机械臂整体视为智能体，采用DDPG算法来训练机械臂，将训练结果作为对比。图10(a)和图10(b)分别为本文所提出方法和仅使用

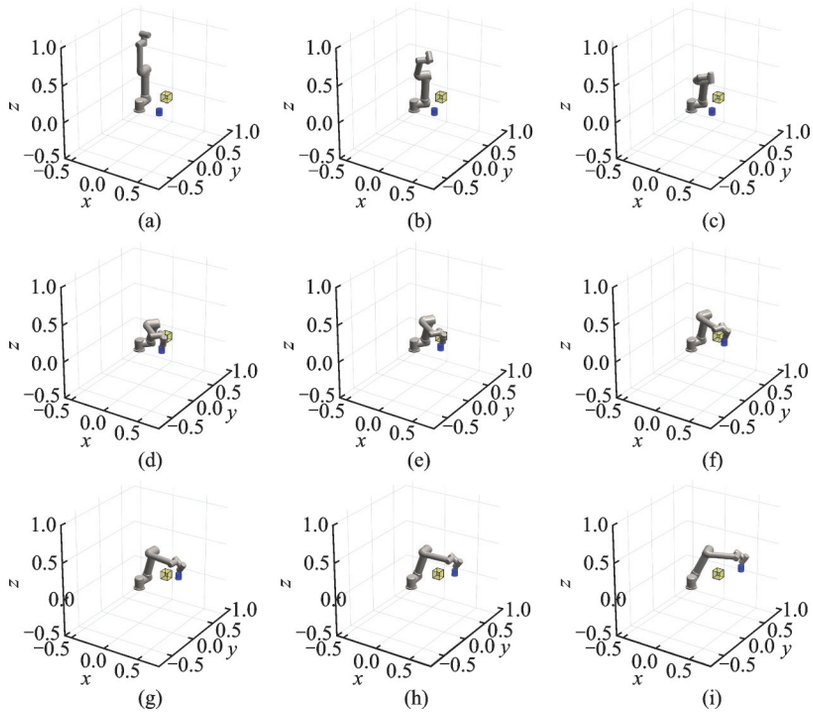


图 7 测试场景一机械臂的状态变化

Fig.7 State of the manipulator changes in scenario 1

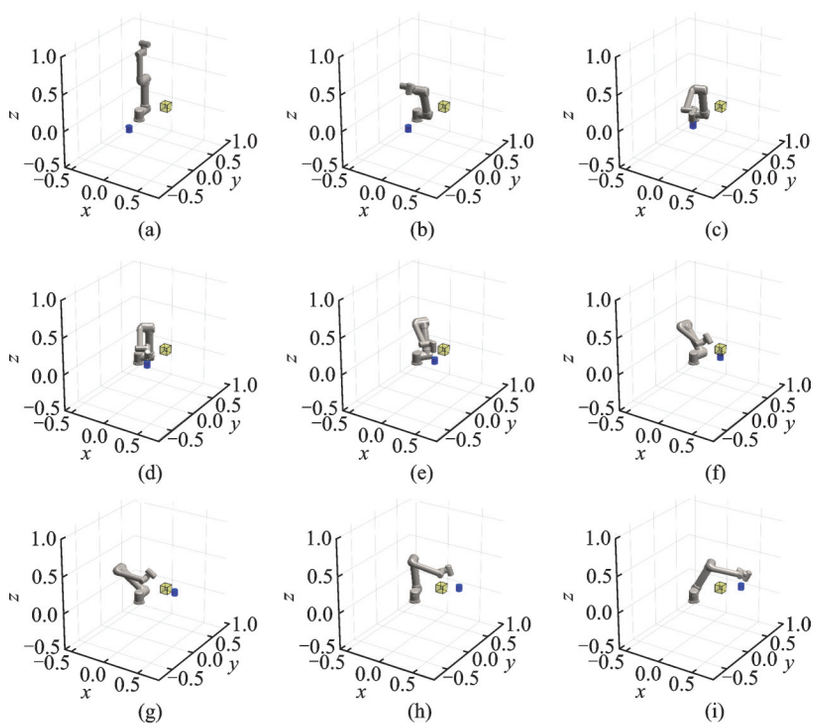
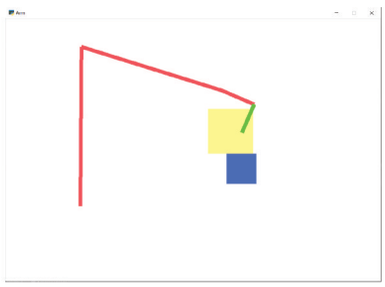


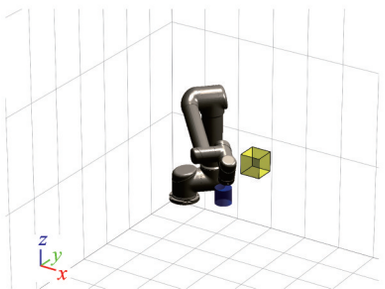
图 8 测试场景二机械臂的状态变化

Fig.8 State of the manipulator changes in scenario 2

DDPG 算法训练过程所获得的奖励情况。因为存在障碍物对机械臂的运动有阻碍和无阻碍两种不同场景，机械臂为避开障碍物而远离目标物从而导致奖励减小，所以奖励会有较大波动。DDPG-PID 方法不仅能够获得更高的奖励，而且收敛性要比仅使用 DDPG 算法好，前者奖励在后期能较为稳定地收敛并获得 1 000 上下浮动的奖励，机械臂能够获得较为理想的自主跟踪目标与避障的能力；后者奖励在后期出现下降的趋势，机械臂自主决策能力（目标跟踪与避障）不足。这表明，DDPG-PID 控制能够有效确保机械臂的控制效果，实现对目标稳定跟踪的同时避开障碍物。



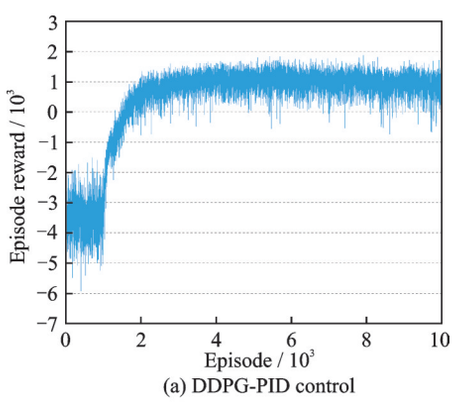
(a) State of the manipulator on the working plane



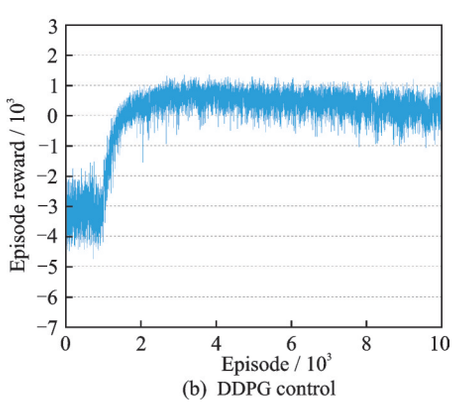
(b) State of the manipulator in three-dimensional space

图 9　测试场景二机械臂典型状态

Fig.9　Typical state of the manipulator in scenario 2



(a) DDPG-PID control



(b) DDPG control

图 10　训练过程的奖励情况

Fig.10　Reward during training

5　结　　论

本文将传统PID控制和DDPG强化学习结合，提出了DDPG-PID控制方法控制机械臂在动态复杂环境下实现对目标的稳定跟踪和避障。引入PID控制使得算法收敛效果和控制性能更好。关节偏置的引入使得模型更加贴合实际机械臂的构型，而避障纠正因子的引入成功解决了投影法会造成机械臂对碰撞误判的问题。本文后续将针对机械臂对动态目标的跟踪精度问题以及机械臂操作环境存在动态障碍物的情况，结合模型预测控制进行进一步研究。

参考文献：

[1] SICILIANO B, SCIAVICCO L, VILLANI L, et al. Robotics: Modelling, planning and control[M]. London: Springer-Verlag, 2010.

[2] 祝敬，杨马英. 基于改进人工势场法的机械臂避障路径规划[J]. 计算机测量与控制，2018，26(10)：205-210.
ZHU Jing, YANG Maying. Path planning of manipulator to avoid obstacle based on improved artifical potential field method[J]. Computer Measurement & Control, 2018, 26(10): 205-210.

[3] 马宇豪，梁雁冰. 一种基于六次多项式轨迹规划的机械臂避障算法[J]. 西北工业大学学报，2020，38(2)：392-400.
MA Yuhao, LIANG Yanbing. An obstacle avoidance algorithm for manipulators based on six-order polynomial trajectory planning[J]. Journal of Northwestern Polytechnical University, 2020, 38(2): 392-400.

[4] WANG M, LUO J, WALTER U. A non-linear model predictive controller with obstacle avoidance for a space robot[J]. Advances in Space Research, 2016, 57(8): 1737-1746.

[5] JORDAN M I, MITCHELL T M. Machine learning: Trends, perspectives, and prospects[J]. Science, 2015, 349(6245): 255-260.

[6] SUTTON R S, BARTO A G. Reinforcement learning: An introduction[M]. 2nd ed. Cambridge, MA: MIT Press, 2018.

[7] 刘乃军，鲁涛，蔡莹皓，等. 机器人操作技能学习方法综述[J]. 自动化学报，2019，45(3)：458-470.
LIU Naijun, LU Tao, CAI Yinghao, et al. A review of robot manipulation skills learning methods[J]. Acta Automatica Sinica, 2019, 45(3): 458-470.

[8] KOBER J, BAGNELL J A, PETERS J. Reinforcement learning in robotics: A survey[J]. International Journal of Robotics Research, 2013, 32(11): 1238-1274.

[9] 李鹤宇，赵志龙，顾蕾，等. 基于深度强化学习的机械臂控制方法[J]. 系统仿真学报，2019，31(11)：2452-2457.

LI Heyu, ZHAO Zhilong, GU Lei, et al. Robot arm control method based on deep reinforcement learning [J]. Journal of System Simulation, 2019, 31(11): 2452-2457.

[10] SARANTOPOULOS I, KIATOS M, DOULGERI Z, et al. Split deep q-learning for robust object singulation[C]//Preceedings of 2020 IEEE International Conference on Robotics and Automation. Piscataway, NJ, USA: IEEE, 2020: 6225-6231.

[11] 徐帷, 卢山. 基于Sarsa(λ)强化学习的空间机械臂路径规划研究[J]. 宇航学报, 2019, 40(4): 435-443.
XU Wei, LU Shan. Analysis of space manipulator route planning based on Sarsa(λ) reinforcement learning[J]. Journal of Astronautics, 2019, 40(4): 435-443.

[12] CHRISTEN S, JENDELE L, AKSAN E, et al. Learning functionally decomposed hierarchies for continuous control tasks with path planning[J]. IEEE Robotics and Automation Letters, 2021, 6(2): 3623-3630.

[13] SANGIOVANNI B, RENDINIELLO A, INCREMONA G P, et al. Deep reinforcement learning for collision avoidance of robotic manipulators[C]//Proceedings of European Control Conference (ECC). New York, USA: IEEE, 2018: 2063-2068.

[14] CHATZILYGEROUDIS K, VASSILIADES V, STULP F, et al. A survey on policy search algorithms for learning robot controllers in a handful of trials[J]. IEEE Transactions on Robotics, 2020, 36(2): 328-347.

[15] ZHONG J, WANG T, CHENG L L. Collision-free path planning for welding manipulator via hybrid algorithm of deep reinforcement learning and inverse kinematics[J]. Complex & Intelligent Systems, 2021. DOI: http://dx.doi.org/10.1007/s40747-021-00366-1.

[16] LIN Y, HUANG J, ZIMMER M, et al. Invariant transform experience replay: Data augmentation for deep reinforcement learning[J]. IEEE Robotics and Automation Letters, 2020, 5(4): 6615-6622.

[17] JOHANNINK T, BAHL S, NAIR A, et al. Residual reinforcement learning for robot control[C]//Proceedings of 2019 International Conference on Robotics and Automation. New York, USA: IEEE, 2019: 6023-6029.

[18] YAMADA J, LEE Y, SALHOTRA G, et al. Motion planner augmented reinforcement learning for robot manipulation in obstructed environments[J]. arXiv pre-print server, 2020. DOI: https://arxiv.org/abs/2010.11940.

[19] AL-GABALAWY M. A hybrid MPC for constrained deep reinforcement learning applied for planar robotic arm[J]. ISA Transactions, 2021. DOI: 10.1016/j.isatra.2021.03.046.

[20] KUKKER A, SHARMA R. Stochastic genetic algorithm-assisted fuzzy q-learning for robotic manipulators[J]. Arabian Journal for Science and Engineering, 2021. DOI: 10.1007/s13369-021-05379-z.

[21] OTA K, JHA D K, OIKI T, et al. Trajectory optimization for unknown constrained systems using reinforcement learning[C]//Proceedings of 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems. New York, USA: IEEE, 2019: 3487-3494.

(编辑:夏道家)