DOI: 10.13382/j.jemi.B2407836

多智能体深度强化学习优化的机器人导纳控制*

李逃昌[1] 李健璋[1] 侯利民[1] 金海波[2]
(1. 辽宁工程技术大学电气与控制工程学院 葫芦岛 125105;2. 辽宁工程技术大学软件学院 葫芦岛 125105)

摘 要:针对固定参数主动柔顺控制受机器人内部参数不确定等建模误差影响导致轨迹精度不高的问题,提出一种基于多智能体深度确定性策略梯度(MA-DDPG)的机器人自适应导纳控制方法。首先,基于机器人模型建立导纳控制器。其次,将深度确定性策略梯度(DDPG)算法与导纳控制相结合,设计了一种由 DDPG 智能体直接输出导纳参数的自适应导纳控制器。针对其收敛速度慢和控制效果不好的问题,在自适应导纳控制算法中引入多智能体思想,将每一个导纳控制参数作为一个智能体的输出,采用集中式训练分布式执行架构的 MA-DDPG 算法对导纳控制器参数进行协同优化。最后,通过对比深度强化学习仿真训练效果以及自适应导纳控制在期望轨迹上的受力实验效果,验证了所提方法的可行性与有效性。实验数据表明,与其他深度强化学习算法的自适应导纳控制相比,所提方法的仿真训练收敛速度提高了 65.88%,轨迹精度提高了 63.35%。

关键词: 机器人;深度强化学习;导纳控制
中图分类号: TP242;TN06 **文献标识码**: A **国家标准学科分类代码**: 510.80

Robot admittance control optimized by multi-agent deep reinforcement learning

Li Taochang[1] Li Jianzhang[1] Hou Limin[1] Jin Haibo[2]
(1. Faculty of Electrical and Control Engineering, Liaoning Technical University, Huludao 125105, China;
2. Faculty of Software, Liaoning Technical University, Huludao 125105, China)

Abstract: The paper proposes a robot adaptive admittance control method based on the multi-agent deep deterministic policy gradient (MA-DDPG) to address the issue of low trajectory accuracy in fixed-parameter active compliance control caused by modeling errors, such as uncertainty in robot internal parameters. Firstly, an admittance controller is established based on the robot model. Secondly, by integrating the DDPG algorithm with the admittance control framework, an adaptive admittance controller is developed, wherein the DDPG-based agent dynamically generates optimal admittance parameters. To address issues of slow convergence and unsatisfactory control performance, the concept of multiple agents is introduced into the adaptive admittance control algorithm, with each agent responsible for optimizing an individual admittance control parameter. The MA-DDPG algorithm, based on a centralized training and distributed execution architecture, is employed to optimize the admittance controller parameters. Finally, the feasibility and effectiveness of the proposed method are validated through a comparative analysis between the impact of deep reinforcement learning simulation training and the experimental outcomes of adaptive admittance control on the anticipated trajectory. The experimental data demonstrate that in comparison with adaptive admittance control based on alternative deep reinforcement learning algorithms, the proposed method exhibits a 65.88% improvement in convergence speed and a 63.35% enhancement in trajectory accuracy.

Keywords: robot; deep reinforcement learning; admittance control

收稿日期:2024-09-19 Received Date: 2024-09-19
*基金项目:国家自然科学基金(52177047,62173171)、葫芦岛科技计划项目(2024JH(2)2/05b)资助

0　引　言

近年来,随着新技术的巨大进步,机器人的应用也日益广泛,已延伸到了人们生活和生产的各个领域[1]。与环境进行明确定义的物理交互是机器人领域大量实际应用的基本要求[2],用来帮助和代替人类工作的机器人技术在各个领域都取得了显著进展[3]。然而,随着机器人技术的不断进步,对机器人在柔顺控制方面的要求也越来越高,柔顺控制对于提高机器人的作业精度和安全性至关重要。

柔顺控制分为被动柔顺控制和主动柔顺控制,被动柔顺控制由于依靠弹性元件,无法从根本上解决精度与柔顺性之间的矛盾,因此存在适应性差、精度低的缺点;主动柔顺控制包括阻抗控制[4]和导纳控制[5]。其中以运动学参数为输入,以关节转矩为输出的阻抗控制适合于大刚度环境,而以交互力为输入,以运动学参数为输出的导纳控制更适合于人机交互过程[6],其实这两种机器人柔顺控制策略的本质是一样的。面对更复杂的环境和更深层次的人机交互时,导纳/阻抗控制的一个主要挑战是控制器的参数设置为常数,无法适应不同的交互要求和非结构化环境[7]。为此,越来越多的研究学者注意到了可变参数的主动柔顺控制的优势。文献[8]利用自抗扰控制器作为内环控制,用粒子群神经网络作为外环控制,在线优化导纳/阻抗参数解决自适应调节能力不足问题;文献[9]通过滑模控制与阻抗控制的结合,设计非奇异快速终端滑模控制方法来解决不确定动力学和外部干扰的影响,实现假肢腿机器人的路径跟踪摆动和姿态控制;文献[10]利用卷积神经网络估计人类的意图,设计自适应模糊控制器在每次迭代中估计导纳/阻抗控制参数解决用户与其轨迹的不确定性。虽然与控制算法的结合能够一定程度上提高控制性能,但控制算法需要精确的系统建模,建模误差会显著影响控制效果,且正逆运动学的误差还会叠加,另外在处理高度非线性的复杂任务时控制算法表现不佳。

强化学习(reinforcement learning, RL)是通过与环境交互进行学习,使智能体能够长期最大化累计奖励的一种机器学习(machine learning, ML)算法[11]。随着机器学习算法的快速发展与应用,机器人导纳/阻抗控制策略与机器学习算法的结合得到了进一步发展[12]。强化学习算法只需通过不断尝试与环境交互,就能积累经验,依据评价指导智能体做出最优决策[13]。随着计算机算力的提高,结合了深度学习的深度强化学习(deep reinforcement learning, DRL)具备了解决复杂问题的能力[14],并在机器人领域得到广泛应用。例如应用于机械臂仿人运动规划[15]和移动机器人路径规划[16-17]以及轨迹跟踪[18],这为机器人实现复杂的主动柔顺控制提供了新途径。在基于强化学习实现可变参数导纳/阻抗控制的研究中,文献[19]利用神经网络补偿系统不确定性和外部干扰,通过建立导纳/阻抗误差提出自适应导纳/阻抗控制器,提高系统的控制性能;文献[20]提出了一种用于人机物理协作的自适应导纳/阻抗控制策略。利用 Q 学习估计人的控制行为来寻找最优导纳/阻抗参数集,实时自适应地调节机器人控制的运动参考和导纳/阻抗参数,减小跟踪误差;文献[21]以机械臂状态作为输入,使用近端策略优化(proximal policy optimization, PPO)输出跟踪半径和导纳/阻抗参数,实现机械臂的恒力跟踪,但 PPO 这种输出概率性策略的多智能体深度强化学习算法无法高效地处理连续动作空间,利用策略参数化处理连续动作空间会增加计算复杂性。以上研究表明在机器人主动柔顺控制系统中引入深度强化学习机制虽然可以有效避免控制器参数设置的不合理性,但是这种方法对于多个指标的控制系统在奖励函数设计上依然面临求最优参数问题,不同奖励函数的权重设置不合理会直接影响算法的收敛性。

综上所述,本文针对固定参数主动柔顺控制受机器人内部参数不确定等建模误差影响导致轨迹精度不高的问题,提出一种基于多智能体深度强化学习的机器人自适应导纳控制方法。首先,利用深度强化学习智能体与导纳控制相结合来实现机器人自适应导纳控制。然后在此基础上,本文创新性地将多智能体思想引入到机器人导纳控制算法中,利用相互通信的两个智能体自适应调节导纳控制参数,减小了机器人末端轨迹误差的同时也加快深度强化学习参数寻优的收敛速度。最后,通过仿真训练和对比实验,验证所提方法的可行性和有效性。

1　自适应导纳控制

1.1　导纳控制

根据导纳控制算法原理,当存在外部力时,导纳控制器的输出会顺应外部力进行修正,以模拟弹簧阻尼系统,实现柔顺控制。机器人笛卡尔空间下导纳控制器为:

$$\ddot{\tilde{\boldsymbol{x}}} = \boldsymbol{M}^{-1}(\boldsymbol{F}_e - \boldsymbol{B}\dot{\tilde{\boldsymbol{x}}} - \boldsymbol{K}\tilde{\boldsymbol{x}}) \tag{1}$$

式中:$\boldsymbol{M}$、$\boldsymbol{B}$、$\boldsymbol{K}$ 分别表示目标惯性对角阵、目标阻尼对角阵以及目标刚度对角阵;$\boldsymbol{F}_e$ 为笛卡尔坐标系下的末端外部力;$\tilde{\boldsymbol{x}}$、$\dot{\tilde{\boldsymbol{x}}}$ 分别为实际末端位姿与目标位姿的位置偏差和速度偏差;$\ddot{\tilde{\boldsymbol{x}}}$ 为期望的加速度偏移量。机器人导纳控制系统如图 1 所示,其中,$\boldsymbol{x}_d$、$\dot{\boldsymbol{x}}_d$ 分别为期望的目标位置和目标速度;$\boldsymbol{x}_r$、$\dot{\boldsymbol{x}}_r$ 分别为实际的末端位置和末端速度;$\boldsymbol{x}_e$ 为导纳控制器期望的位置偏差;$\boldsymbol{x}_c$ 为导纳控制器

期望的目标位置；$\boldsymbol{q}_c$、$\boldsymbol{q}_r$、$\dot{\boldsymbol{q}}_r$ 分别为期望的角度输入量、机器人反馈的实际角度和角速度。图1中的运动学部分由于求解算法的近似和模型的不准确，会出现误差，而且正运动学和逆运动学的误差会相互叠加，影响机器人末端执行器的精度。为减小误差又保留系统的柔顺性，需要系统自适应地调节导纳参数，但又因为误差计算复杂且关系难以确定，因此引入深度强化学习进行参数优化调节。

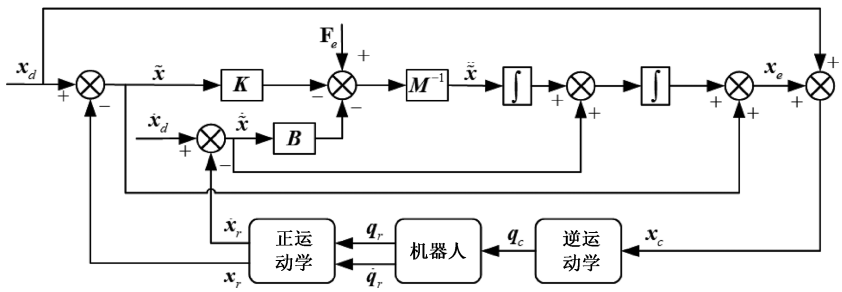

图1　导纳控制器结构框图

Fig. 1　Admittance controller structure block diagram

1.2　深度确定性策略梯度算法

1）马尔可夫决策过程

深度强化学习是一种结合了深度学习和强化学习的技术，在近年来取得了许多突破性进展，解决了传统强化学习难以处理的复杂问题。强化学习的核心思想是通过智能体与环境的持续交互，智能体在不同的情境下采取不同的行动序列，并根据环境对这些行动的反馈信号进行评估。智能体根据这些反馈来调整和优化未来的行为。具体来说就是当智能体的某个行为在环境中获得了正向回报，该行为在相同环境条件下被再次采用的可能性就会增加；反之，如果行为得到负向回报，该行为被再次采用的可能性则会减少。通过不断尝试和调整，智能体最终可以学习到一组最优策略。这一过程实际上是一个复杂的非线性高维映射，即从动作到环境反应的关系映射。

机器人导纳控制与深度强化学习结合的首要任务就是建立马尔可夫决策过程（Markov decision process，MDP）模型[22]。通常 MDP 模型由状态空间 S、动作空间 A、状态转移概率 P、奖励函数 R 与折扣因子 γ 这5个元素构成。首先选择机器人的关节角度与关节速度作为系统状态，所以状态 $\boldsymbol{s}$ 是一个12维特征向量：

$$\boldsymbol{s}=[\boldsymbol{q},\dot{\boldsymbol{q}}] \tag{2}$$

式中：$\boldsymbol{q}$、$\dot{\boldsymbol{q}}$ 分别为机器人关节角度和关节角速度；动作空间 A 为具体的导纳参数集；转移概率 P 取决于执行动作后系统的反馈，需要依靠多次蒙特卡洛采样对其进行无偏估计；周期累计奖励函数 R 设计如式(3)所示。

$$R=\sum_{t=1}^{N}\gamma^t r^t=-\sum_{t=1}^{N}\gamma^t\left|\boldsymbol{x}_d^t-\boldsymbol{x}_r^t\right| \tag{3}$$

式中：r^t 为当前步 t 的奖励函数；γ^t 为当前步的折扣因子，表示智能体越快的找到最佳策略，其得到的奖励就越高；$\boldsymbol{x}_r^t$、$\boldsymbol{x}_d^t$ 分别为当前步机器人末端的实际位置和期望轨迹位置。因为要尽可能的减小误差，所以在绝对值外取负，越接近0表示误差越小。

2）确定性策略梯度和 Actor-Critic

Q-Learning 是强化学习中最经典的方法，通过学习 Q 值函数间接选择最优动作，递归形式的 Q 函数（基于策略 π）可写为：

$$Q^{\pi}(\boldsymbol{s},\boldsymbol{a})=\mathrm{E}_{s'}[r(\boldsymbol{s},\boldsymbol{a})+\gamma\mathrm{E}_{a'\pi}[Q^{\pi}(\boldsymbol{s}',\boldsymbol{a}')]] \tag{4}$$

其计算了下一状态 $\boldsymbol{s}'$ 以后所有的累计奖励。但因为 Q-Learning 不能处理高维度观测空间问题，因此诞生了 DQN（deep Q network）算法。

DQN 是 Q-Learning 的扩展，使用深度神经网络处理高维状态空间，通过最小化损失函数来学习最优策略对应的动作值函数。通过包含元组 $(\boldsymbol{s},\boldsymbol{a},r,\boldsymbol{s}')$ 的经验重放缓冲区 D 来使 DQN 能够稳定。但 Q-Learning 和 DQN 都只能处理离散的、低维的动作空间，为了处理连续的、高维的动作空间且保证充分的探索，文献[23]提出了异策略确定性策略梯度（deterministic policy gradient，DPG）。

DPG 直接优化确定性策略，采用演员-评论家（actor-critic，AC）架构，AC 架构框图如图2所示，能够从每一步经验中提取更多的信息，从而提高样本效率。在处理连续动作空间时，AC 架构特别有效。演员网络直接输出具体的动作值，而评论家网络则评估这些动作的好坏，避免了在连续动作空间中进行动作选择的复杂性。

由图2可知，Actor 网络负责生成给定状态下的确定性动作。其目标是生成能够最大化预期回报的动作。使用评论家（Critic）网络估计的 Q 值来指导策略网络的更新。通过最大化 Q 值来更新策略网络的参数，能够更稳定地更新和更适应连续动作空间。尽管如此，由于存在 Actor 网络参数化问题和 Critic 网络维数灾难问题，采用

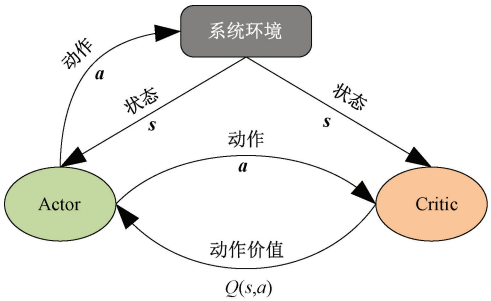

图 2　AC 架构框图

Fig. 2　AC architecture block diagram

AC 架构的确定性策略梯度在面对困难问题时依然是不稳定的,为了解决此问题,文献[24]将 AC 方法与 DQN 结合起来提出了深度确定性策略梯度(deep deterministic policy gradient,DDPG)算法。

3) DDPG

DDPG 算法融合了深度 Q 网络和确定性策略梯度的思想,通过使用神经网络来逼近策略和 Q 值函数,能够有效地处理高维连续动作空间问题,输出确定性动作。实现了在复杂环境中的高效学习。由于同一个网络参数在频繁地进行梯度更新和梯度计算会造成学习过程不稳定的问题。DDPG 的解决方案是把 Actor 网络和 Critic 网络都分为在线(online)网络和目标(target)网络,在这里动作设计为 Actor 目标网络输出的导纳参数,即当前步的动作为:

$$\boldsymbol{a}_t = \mu'(\boldsymbol{s}_t \mid \theta_{\mu'}) = \mu(\boldsymbol{s}_t \mid \theta_\mu) + \boldsymbol{N} = \begin{bmatrix} \boldsymbol{B}_t \\ \boldsymbol{K}_t \end{bmatrix} \tag{5}$$

式中:μ' 为 Actor 目标网络;$\theta_{\mu'}$ 为其网络参数;同理 μ、θ_μ 分别为 Actor 在线网络及其参数;$\boldsymbol{N}$ 为噪声用于促进探索,避免陷入局部最优;$\boldsymbol{B}_t$、$\boldsymbol{K}_t$ 为当前状态的导纳参数。Critic 网络中对参数 θ 的更新采用 DQN 中的 TD 误差方式,当前步的 TD 目标函数为:

$$y = r + \gamma \max_{a'} Q'(\boldsymbol{s}', \boldsymbol{a}' \mid \theta_{Q'}) \tag{6}$$

式中:$\boldsymbol{s}'$、a' 分别为下一时间步的状态和动作;Q'、$\theta_{Q'}$ 为 Critic 目标网络及其参数;Critic 在线网络的损失函数为均方误差:

$$L(\theta_Q) = \mathrm{E}_{(\boldsymbol{s},\boldsymbol{a},r,\boldsymbol{s}')\sim D}[(y - Q(\boldsymbol{s},\boldsymbol{a} \mid \theta_Q))^2] \tag{7}$$

Critic 在线网络通过最小化损失函数来更新参数 θ_Q,Actor 在线网络通过最大化 Q 函数的预期值进行更新,其目标函数为:

$$J(\theta_\mu) = \mathrm{E}_{s\sim D}[Q(\boldsymbol{s}, \mu(\boldsymbol{s} \mid \theta_\mu) \mid \theta_Q)] \tag{8}$$

根据梯度下降法,更新梯度可以写成:

$$\nabla_{\theta_\mu} J(\theta_\mu) = E_{s\sim D}[\nabla_{\mu(s)} Q(\boldsymbol{s}, \mu(\boldsymbol{s} \mid \theta_\mu) \mid \theta_Q) \nabla_{\theta_\mu} \mu(\boldsymbol{s} \mid \theta_\mu)] \tag{9}$$

DDPG 算法训练流程如算法 1 所示。

算法 1:DDPG 算法

1. 随机初始化 Critic 和 Actor 在线网络参数 θ_Q 和 θ_μ
2. 初始化 Critic 和 Actor 目标网络参数 $\theta_{Q'} \leftarrow \theta_Q, \theta_{\mu'} \leftarrow \theta_\mu$
3. 初始化经验区 D
4. 初始化一个随机过程 $\boldsymbol{N}$ 用来给动作添加噪声
5. 获取初始化状态 $\boldsymbol{s}_1$
6. **for** $t = 1, \cdots, T$
7. 根据当前策略和探索噪声获得动作 $a_t = \mu(\boldsymbol{s}_t \mid \theta_\mu) + \boldsymbol{N}$
8. 通过执行动作 $\boldsymbol{a}_t$ 获得奖励 r_t 和下一状态 $\boldsymbol{s}_{t+1}$
9. 将 $(\boldsymbol{s}_t, \boldsymbol{a}_t, r_t, \boldsymbol{s}_{t+1})$ 存储到经验区 D 中
10. 从经验区中随机采样 M 个元组 $(\boldsymbol{s}_i, \boldsymbol{a}_i, r_i, \boldsymbol{s}_{i+1})$
11. $y_i = r_i + \gamma Q'_i(\boldsymbol{s}_{i+1}, \mu'(\boldsymbol{s}_{i+1} \mid \theta_{\mu'}) \mid \theta_{Q'})$
12. 通过最小化损失函数更新 Critic 在线网络参数 θ_Q

$$L = \frac{1}{2M} \sum_{i=1}^{M} (y_i - Q(\boldsymbol{s}_i, \boldsymbol{a}_i \mid \theta_Q))^2$$

13. 通过计算策略梯度更新 Actor 在线网络参数 θ_μ

$$\nabla_{\theta_\mu} J \approx \frac{1}{M} \sum_{i=1}^{M} \nabla_{\mu(s_i)} Q(\boldsymbol{s}_i, \mu(\boldsymbol{s}_i)) \nabla_{\theta_\mu} \mu(\boldsymbol{s}_i)$$

14. 通过滑动平均更新 Critic 和 Actor 目标网络参数

$$\theta_{Q'} \leftarrow \tau \theta_Q + (1 - \tau) \theta_{Q'}$$

$$\theta_{\mu'} \leftarrow \tau \theta_\mu + (1 - \tau) \theta_{\mu'}$$

15. end for

在 DDPG 通过梯度更新在线(online)网络参数后,再通过滑动平均方法更新目标(target)网络参数:

$$\theta_{Q'} \leftarrow \tau \theta_Q + (1 - \tau) \theta_{Q'} \tag{10}$$

$$\theta_{\mu'} \leftarrow \tau \theta_\mu + (1 - \tau) \theta_{\mu'} \tag{11}$$

式中:τ 是一个值远小于 1 的平滑因子,这表示目标网络参数只能缓慢变化,极大的提高了学习的稳定性。

尽管近年来,DDPG 一直是控制领域最先进的深度强化学习方法之一,但上述的训练效果却并不理想,原因是单一的奖励无法收敛多个控制参数,即便将多个指标都引入奖励函数,但仍无法确定每个控制参数对结果影响的权重,为解决这个问题,引入多智能体思想,采用多智能体深度确定性策略梯度(multi-agent deep deterministic policy gradient,MA-DDPG)算法。

2　基于 MA-DDPG 的导纳控制

综上,本文提出一种基于 MA-DDPG 的导纳控制策略,在该策略中,导纳控制的 2 个参数分别采用 2 个智能体(Agent)进行自适应调节,每个智能体通过 DDPG 实现,进而得到基于 2 个智能体协同的导纳参数调节策略,如图 3 所示。因为 MA-DDPG 依然是确定性策略,所以动作依然设计为输出确定的控制参数:

$$\boldsymbol{a}_t^1 = \mu'_1(\boldsymbol{s}_t^1 \mid \theta_{\mu'_1}) = \mu_1(\boldsymbol{s}_t^1 \mid \theta_{\mu_1}) + \boldsymbol{N}_1 = [\boldsymbol{B}_t] \tag{12}$$

式中:$\boldsymbol{a}_t^1$、$\boldsymbol{s}_t^1$ 分别为智能体 1(Agent1)在 t 时间步的动作

和状态；μ'_1、μ_1 分别为智能体1的Actor目标网络和在线网络；N_1 为智能体1的动作噪声，同理智能体2(Agent2)的动作为：

$$a_t^2 = \mu'_2(s_t^2 \mid \theta_{\mu'_2}) = \mu_2(s_t^2 \mid \theta_{\mu_2}) + N_2 = [K_t] \quad (13)$$

如果智能体仅仅知道自己的观测和动作是不足以评价其当前动作是否正确，因此状态被设计为：

$$s_t^1 = [q, \dot{q}, a_{t-1}^2] \quad (14)$$

$$s_t^2 = [q, \dot{q}, a_{t-1}^1] \quad (15)$$

对于初始值 a_0^1 和 a_0^2 选取一组相对稳定的导纳参数，这并不会影响后面的网络训练。每一个智能体都采用DDPG实现，MA-DDPG的输出代入式(1)中，便构成了机器人自适应导纳控制系统，具体如图3所示。

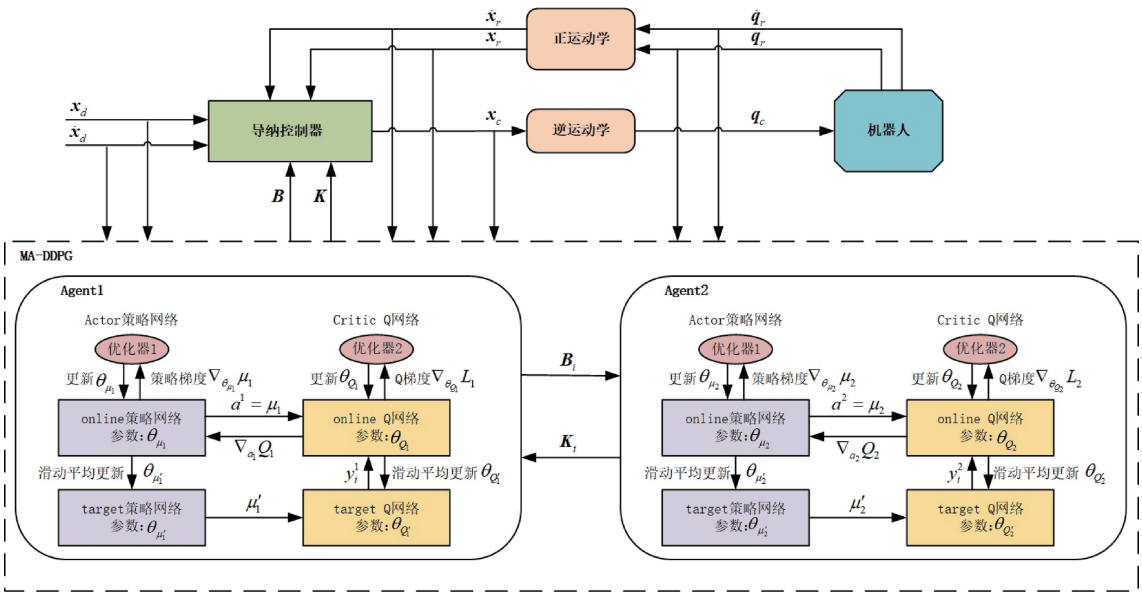

图3　基于MA-DDPG的机器人自适应导纳控制系统框图

Fig. 3　Block diagram of admittance control system based on MA-DDPG

另外神经网络能否收敛很大程度上取决于奖励函数是否合理，所以奖励函数的设计至关重要，它会直接影响算法的学习效果。如果阻尼系数过大，受力偏移后很难回到轨迹上，导致误差增大，而过小会导致机器人末端不稳定，出现振荡同样会增大轨迹误差，因此Agent1(进行阻尼参数调节)的奖励函数设计如(16)所示，MA-DDPG的算法流程如算法2所示。

$$r_t^1 = -\left|x_d^t - x_r^t\right| \quad (16)$$

但如果使用轨迹误差来训练刚度系数，会导致刚度无限大，最终机器人为了减少轨迹误差会失去柔顺性，而本文想要的是在保留柔顺性的基础上减小误差，所以采用曲率作为刚度系数的奖励函数，但由于曲率的计算需要用到曲线拟合，大量的计算会加长算法的学习周期，大大增加了训练时间，因此本文基于轨迹点连线的夹角进行Agent2(进行刚度参数调节)的奖励函数的设计，如式(17)所示。

$$r_t^2 = -\arccos\left(\frac{\overrightarrow{P_{t-2}P_{t-1}} \cdot \overrightarrow{P_{t-2}P_t}}{\left|\overrightarrow{P_{t-2}P_{t-1}}\right| \cdot \left|\overrightarrow{P_{t-2}P_t}\right|}\right) \quad (17)$$

式中：P_t 为 t 时间步机器人实际的末端位置；$\overrightarrow{P_{t-2}P_t}$ 为点 P_{t-2} 到点 P_t 的向量；$|\overrightarrow{P_{t-2}P_t}|$ 为向量的范数(长度)。考虑实际轨迹的稳定性和机器人初始姿态对训练的影响，算法的训练由第2个周期开始，且机器人的期望轨迹是一个封闭圆，所以在初始位置当 $t=1$ 时，P_{t-2} 和 P_{t-1} 为上一个周期的最后两个点。

3　验证与分析

3.1　MA-DDPG算法仿真训练

本文算法的训练设备处理器为英特尔Xeon(至强)Gold 6248R 3.00 GHz(X2)。采用MATLAB/Simulink中的VR robot构建深度强化学习仿真环境，用于算法测试和神经网络参数的训练。MA-DDPG算法的超参数如表1所示。

目前领域内公认的基准算法有DDPG算法、双延迟深度确定性策略梯度(twin delayed deep deterministic policy gradient，TD3)算法[25]和“软演员-评论家”(soft actor-critic，SAC)算法[26]，本文分别采用了上述3种算法作为基准进行对比实验。上述3种单智能体算法的超参数如表2所示。

算法 2:MA-DDPG 算法

1. 随机初始化 Critic 和 Actor 在线网络参数 θ_{Q_1}、θ_{Q_2}、θ_{μ_1}、θ_{μ_2}
2. 初始化 Critic 和 Actor 目标网络参数
   $\theta_{Q'_1} \leftarrow \theta_{Q_1}$、$\theta_{\mu'_1} \leftarrow \theta_{\mu_1}$、$\theta_{Q'_2} \leftarrow \theta_{Q_2}$、$\theta_{\mu'_2} \leftarrow \theta_{\mu_2}$
3. 初始化 Agent1 和 Agent2 的经验区 D_1 和 D_2
4. **for** $j=1,2$ (Agent1,Agent2)
5. 初始化随机过程 $\boldsymbol{N}_j$ 用来给动作添加噪声
6. 获取初始化状态 $\boldsymbol{s}_1^j$
7. **for** $t=1,\cdots,T$
8. 获得当前步动作 $\boldsymbol{a}_t^j = \mu_j(\boldsymbol{s}_t^j \mid \theta_{\mu_j}) + \boldsymbol{N}_j$
9. 执行动作 $\boldsymbol{a}_t^j$ 获得对应的奖励 r_t^j 和下一状态 $\boldsymbol{s}_{t+1}^j$
10. 将元组 $(\boldsymbol{s}_t^j,\boldsymbol{a}_t^j,r_t^j,\boldsymbol{s}_{t+1}^j)$ 存储到经验区 D_j 中
11. 从经验区中随机采样 M 个元组 $(\boldsymbol{s}_i^j,\boldsymbol{a}_i^j,r_i^j,\boldsymbol{s}_{i+1}^j)$
12. $y_i^j = r_i^j + \gamma^j Q'_j(\boldsymbol{s}_{i+1}^j,\mu'_j(\boldsymbol{s}_{i+1}^j \mid \theta_{\mu'_j}) \mid \theta_{Q'_j})$
13. 通过最小化损失函数更新 Critic 在线网络参数 θ_{Q_j}
    $$L_j = \frac{1}{2N}\sum_{i=1}^{N}(y_i^j - Q_j(\boldsymbol{s}_i^j,\boldsymbol{a}_i^j \mid \theta_{Q_j}))^2$$
14. 通过计算策略梯度更新 Actor 在线网络参数 θ_{μ_j}
    $$\nabla_{\theta_\mu} J_j \approx \frac{1}{N}\sum_{i=1}^{N} \nabla_{\mu_j(\boldsymbol{s}_i^j)} Q_j(\boldsymbol{s}_i^j,\mu_j(\boldsymbol{s}_i^j)) \nabla_{\theta_{\mu_j}} \mu_j(\boldsymbol{s}_i^j)$$
15. 通过滑动平均更新 Critic 和 Actor 目标网络参数
    $$\theta_{Q'_j} \leftarrow \tau_j \theta_{Q_j} + (1-\tau_j)\theta_{Q'_j}$$
    $$\theta_{\mu'_j} \leftarrow \tau_j \theta_{\mu_j} + (1-\tau_j)\theta_{\mu'_j}$$
16. **end for**
17. **end for**

表 1　MA-DDPG 算法的超参数

Table 1　Hyperparameters of MA-DDPG algorithm

MA-DDPG	智能体 1	智能体 2
输入层神经元个数	40	36
隐藏层神经元个数	20	18
目标网络平滑因子	0.005	0.005
经验区大小	1 000 000	1 000 000
折扣因子	0.65	0.65
随机采样批量	32	32
噪声标准差	0.2	1.5
噪声衰减率	0.001	0.001
动作 $\boldsymbol{a}_0$ 的取值	diag(1,1,1,1,1,1)	diag(50,50,50,50,50,50)

由表 4 中参数可知,MA-DDPG 算法的神经网络结构与 DDPG、TD3、SAC 算法的神经网络结构近似,因此并没有增加算法的计算复杂度。

VR robot 是一种结合了虚拟现实(VR)技术和机器人仿真技术的工具,是 MATLAB/Simulink Virtual Reality Toolbox 的一部分。它用于创建、模拟和可视化三维虚拟机器人环境,能够在设计和测试机器人系统时进行更直观和互动的仿真。环境中的机器人具有关节和连杆的物理特性以及质量分布和惯性特性。同时可以模拟现实中的摩擦力、阻尼力、重力和外力等。为了使不同算法的训练效果能够进行对比,使用训练中生成的同一外力进行训练,在末端 y 轴的负方向上作用一个大小为 10 N 左右的力,如图 4 所示。

将训练最大回合数设为 2 000,每回合总时间为 20 s,每个轨迹周期为 10 s,在每回合的第 2 周期计算奖励,采样时间设为 0.001 s。对应的深度强化学习算法的训练结果如图 5 所示,各曲线为算法训练过程中的平均累计奖励曲线。图 6 和 7 分别是智能体 1 和 2 的训练效果。

表 2　不同算法的超参数

Table 2　Hyperparameters of different algorithms

DDPG		TD3		SAC	
输入层神经元个数	36	输入层神经元个数	36	输入层神经元个数	32
隐藏层神经元个数	18	隐藏层神经元个数	18	隐藏层神经元个数	16
目标网络平滑因子	0.001	探索噪声标准差	[0.2;1.5]	目标网络平滑因子	0.001
经验区大小	1 000 000	探索噪声衰减率	0.001	经验区大小	1 000 000
折扣因子	0.54	目标平滑噪声标准差	[0.2;1.5]	折扣因子	0.77
随机采样批量	32	目标平滑噪声衰减率	0.001	随机采样批量	32
噪声标准差	[0.2;2]	折扣因子	0.87		
噪声衰减率	0.001	目标网络平滑因子	0.001		
		目标策略平滑模型的方差	0.6		
		经验区大小	1 000 000		

由于奖励机制的不同,每周期累计奖励的值和意义也不一样,因此无法与 DDPG 等算法一同进行对比。

根据图 5 的训练结果可以得出,与输出确定性策略的 DDPG 算法和 TD3 算法相比,采用 AC 架构输出随机策略的 SAC 算法训练过程波动很大,收敛过程不稳定;而 TD3 算法作为 DDPG 的改进算法,在深度确定性策略梯度的基础上采用双评论家(也就是双目标网络和双评估网络)的结构缓解自举,有效的抑制了价值函数过高估

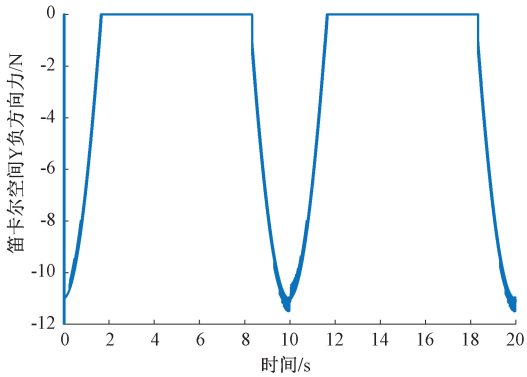

图 4　笛卡尔空间导纳控制器外部力输入

Fig. 4　Cartesian space admittance controller external force input

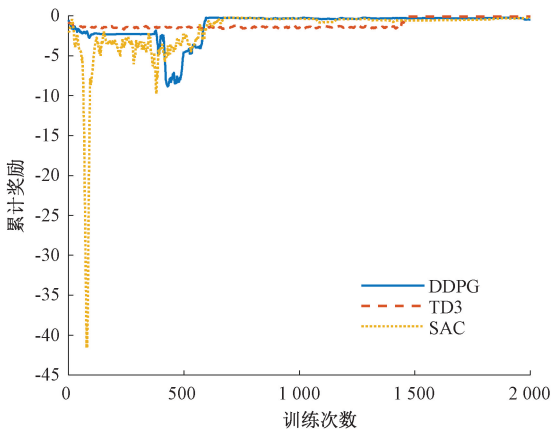

图 5　深度强化学习训练结果

Fig. 5　Training results of deep reinforcement learning

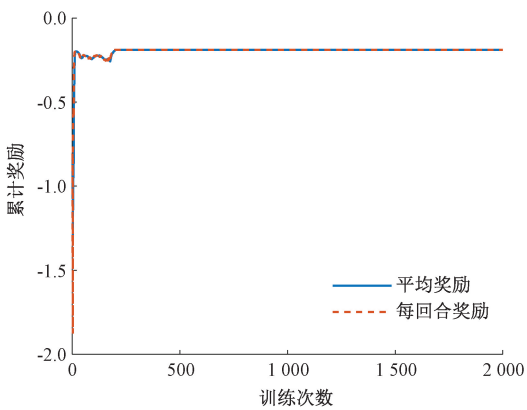

图 6　智能体 1 训练结果

Fig. 6　Training results of agent 1

计问题,训练过程比较稳定,但收敛速度过慢,在 1 500 次左右才收敛,而其他算法在 600 次左右收敛,DDPG 算法收敛最快,在第 595 次后收敛。

由图 6 和 7 可知,采用多智能体思想的 MA-DDPG

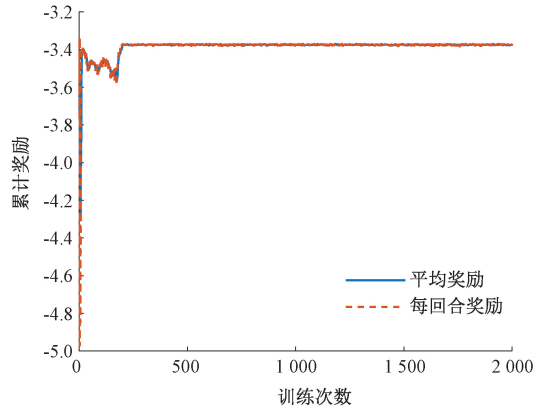

图 7　智能体 2 训练结果

Fig. 7　Training results of agent 2

算法的收敛速度最快,在 203 次左右就已经收敛,且训练过程以及收敛后的输出过程都非常稳定。相比于收敛速度最快的 DDPG 算法,MA-DDPG 算法的收敛速度提高了 65. 88%。

3. 2　实验平台算法验证

实验平台主要分为两部分:导纳参数计算和机械臂运动控制,分别由 PC1 和 PC2 两台电脑完成,通过 UDP 通信协议共同实现机械臂的柔顺控制,如图 8 所示。

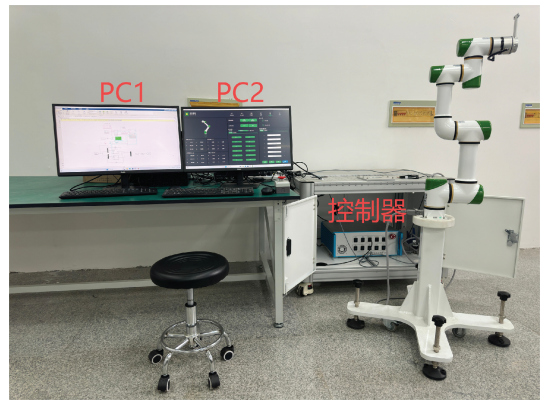

图 8　机械臂导纳控制平台

Fig. 8　Admittance control platform of mechanical arm

采用不同算法时的机器人末端运动轨迹如图 9 所示。

为验证 MA-DDPG 算法的可行性,本文基于机械臂导纳控制平台,针对机械臂运动的柔顺性和轨迹精度,设计了如下验证方案。首先,将设计的期望轨迹以及导纳控制器在 PC2 上利用 MATLAB 自动代码生成功能,编译生成可执行代码,并下发到机械臂的控制器(基于 ARM 的嵌入式平台),进而实现机械臂的运动控制,再通过 PC2 上的示教器实时查看机械臂的状态和参数。然后机械臂控制器会在运行的同时将自身关节状态使用 UDP

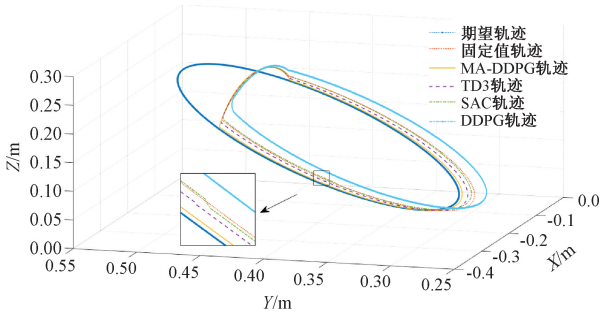

图 9　不同算法的运动轨迹

Fig. 9　Motion trajectories of different algorithms

通信发给上位机 PC1,收到关节状态后 PC1 会把参数调节智能体的输出下发给机械臂控制器,实现自适应导纳控制。

不同轨迹的误差曲线如图 10 所示。由图 10 可以更直观地得出,单智能体的 DDPG 算法效果最差,误差为 0.02 m,固定值导纳控制的误差为 0.01 m,虽然 TD3 算法的轨迹误差相对较小,但可以看出由于输出的导纳参数突变或者参数不合理等原因,出现了一些震荡点,而输出随机策略的 SAC 算法振荡幅度更大,MA-DDPG 的轨迹误差最小也最稳定,误差仅为 0.002 47 m。与误差最小为 0.006 74 m 的 TD3 算法相比,跟踪精度提高了 63.55%。

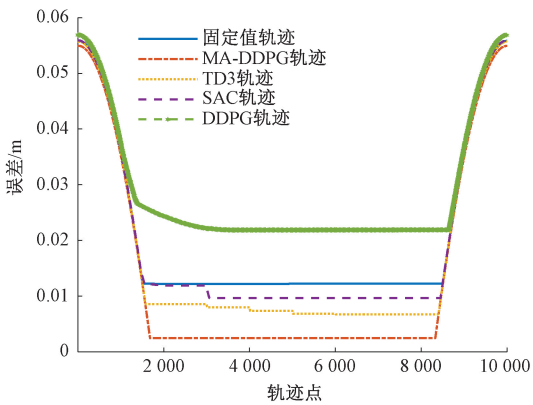

图 10　不同算法轨迹的误差曲线

Fig. 10　Error curves of different algorithm trajectories

4　结　论

本文将多智能体深度强化学习方法创新性地应用于机器人的导纳控制系统中,提出一种自适应的机器人柔顺控制策略。在对导纳控制和深度强化学习深入剖析的基础上,本文构建了基于 MA-DDPG 的自适应导纳控制策略,融合了导纳控制的柔顺性优势与 MA-DDPG 的协同优化能力,实现了导纳控制参数的动态调整与多个指标的协同优化,确保了机器人的最优柔顺控制效果。实验结果表明,本文提出的方法在满足系统柔顺性的前提下,显著提高了奖励函数收敛速度和运动轨迹精度,充分展现了其在控制性能上的优越性。未来将进一步研究该方法的嵌入式部署问题以及开展面向多机械臂协同作业场景的自适应柔顺控制策略研究。

参考文献

[1] PENG G, CHEN C L P, YANG C. Neural networks enhanced optimal admittance control of robot-environment interaction using reinforcement learning [J]. IEEE Transactions on Neural Networks and Learning Systems, 2021, 33(9): 4551-4561.

[2] ISKANDAR M, OTT C, ALBU-SCHÄFFER A, et al. Hybrid force-impedance control for fast end-effector motions[J]. IEEE Robotics and Automation Letters, 2023, 8(7): 3931-3938.

[3] 苏鹏,谢实辉,刘霖,等. 基于阻抗控制的骨外固定机器人矫形力自适应跟踪[J]. 仪器仪表学报,2023,44(11):99-108.
SU P, XIE SH H, LIU L, et al. Based on impedance control bone external fixation orthopaedic robot force adaptive tracking [J]. Chinese Journal of Scientific Instrument, 2023,44(11):99-108.

[4] YE D, YANG C, JIANG Y, et al. Hybrid impedance and admittance control for optimal robot-environment interaction[J]. Robotica, 2024, 42(2): 510-535.

[5] KANG G, Oh H S, SEO J K, et al. Variable admittance control of robot manipulators based on human intention[J]. IEEE/ASME Transactions on Mechatronics, 2019, 24(3): 1023-1032.

[6] 李冬武,张洁,汪俊亮,等. 细纱接头机器人神经网络自适应力跟踪导纳控制[J]. 机械工程学报,2023,59(11):221-231.
LI D W, ZHANG J, WANG J L, et al. Neural network adaptive tracking admittance control for yarn joint robot [J]. Chinese Journal of Mechanical Engineering, 2023, 59(11):221-231.

[7] SUN T, PENG L, CHENG L, et al. Composite learning enhanced robot impedance control [J]. IEEE Transactions on Neural Networks and Learning Systems, 2019, 31(3): 1052-1059.

[8] 郭万金,于苏扬,田玉祥,等. 机器人打磨自适应变阻抗主动柔顺恒力控制[J]. 哈尔滨工业大学学报,2023,55(12):54-65.
GUO W J, YU S Y, TIAN Y X, et al. Adaptive variable impedance active compliant constant force control for

robot grinding [J]. Journal of Harbin Institute of Technology, 2023,55(12):54-65.

[9] JAFARI M, MOBAYEN S, BAYAT F, et al. A nonsingular terminal sliding algorithm for swing and stance control of a prosthetic leg robot [J]. Applied Mathematical Modelling, 2023, 113: 13-29, DOI: 10.1016/j.apm.2022.08.029.

[10] FOROUTANNIA A, AKBARZADEH-T M R, AKBARZADEH A, et al. Adaptive fuzzy impedance control of exoskeleton robots with electromyography-based convolutional neural networks for human intended trajectory estimation[J]. Mechatronics,2023, 91:102952.

[11] 陈亮,梁宸,张景异,等. Actor-Critic 框架下一种基于改进 DDPG 的多智能体强化学习算法[J]. 控制与决策,2021,36(1):75-82.
CHEN L, LIANG CH, ZHANG J Y, et al. A multi-agent reinforcement learning algorithm based on improved DDPG under Actor-Critic framework [J]. Control and Decision-Making, 2021,36(1):75-82.

[12] DING Y, ZHAO J C, MIN X. Impedance control and parameter optimization of surface polishing robot based on reinforcement learning[J]. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 2023, 237(1-2): 216-228.

[13] 闫军威,黄琪,周璇. 基于 DDPG 的冷源系统节能优化控制策略[J]. 控制与决策,2021,36(12):2955-2963.
YAN J W, HUANG Q, ZHOU X. Energy saving optimization control strategy of cold source system based on DDPG [J]. Control and Decision, 2021,36(12): 2955-2963.

[14] 张浩杰,苏治宝,苏波. 基于深度 Q 网络学习的机器人端到端控制方法[J]. 仪器仪表学报,2018,39(10): 36-43.
ZHANG H J, SU ZH B, SU B. End-to-end control method of robot based on deep Q network learning [J]. Chinese Journal of Scientific Instrument, 2018,39(10): 36-43.

[15] 杨傲雷,陈燕玲,徐昱琳. 基于强化学习的机器人手臂仿人运动规划方法[J]. 仪器仪表学报,2021,42(12): 136-145.
YANG AO L, CHEN Y L, XU Y L. Robot arm humanoid motion planning method based on reinforcement learning [J]. Chinese Journal of Scientific Instrument, 2021,42(12):136-145.

[16] 陈际同,周佳加,吴迪,等. 基于 TD3-RRT 的特殊环境下 USV 路径规划算法研究[J/OL]. 系统仿真学报,1-13[2025-04-30]. https://doi.org/10.16182/j.issn1004731x.joss.24-0622.
CHEN J T, ZHOU J J, WU D, et al. Research on USV path planning algorithm based on TD3-RRT [J/OL]. Journal of System Simulation,1-13[2025-04-30]. https://doi.org/10.16182/j.issn1004731x.joss.24-0622.

[17] 王立勇,王弘轩,苏清华,等. 基于改进 Q-Learning 的移动机器人路径规划算法[J]. 电子测量技术,2024,47(9):85-92.
WANG L Y, WANG H X, SU Q H, et al. Path planning algorithm of mobile robot based on improved Q-Learning [J]. Electronic Measurement Technology, 2019, 47(9): 85-92.

[18] 蔡军,苟文耀,刘颜. 基于 actor-critic 框架的在线积分强化学习算法研究[J]. 电子测量与仪器学报,2023,37(3):194-201.
CAI J, GOU W Y, LIU Y. Research on online integral reinforcement learning algorithm based on actor-critic framework [J]. Journal of Electronic Measurement and Instrumentation, 2019,37(3):194-201.

[19] YANG Y, HUANG D, JIN C, et al. Neural learning impedance control of lower limb rehabilitation exoskeleton with flexible joints in the presence of input constraints[J]. International Journal of Robust and Nonlinear Control, 2023, 33(7): 4191-4209.

[20] WU M, HE Y, LIU S. Adaptive impedance control based on reinforcement learning in a human-robot collaboration task with human reference estimation[J]. International Journal of Mechanics and Control,2020,21(1):21-31.

[21] 张思宁. 基于强化学习的协作机器人变阻抗控制方法研究[D]. 哈尔滨:哈尔滨工业大学,2022.
ZHANG S N. Research on variable impedance control method of collaborative robot based on reinforcement learning [D]. Harbin:Harbin Institute of Technology,2022.

[22] 苏永彬,洪瑞康,刘暾东. 基于前馈隐马尔可夫模型的机器人演示轨迹精准重构方法研究[J]. 仪器仪表学报,2023,44(12):199-207.
SU Y B, HONG R K, LIU T D. Research on accurate reconstruction method of robot demonstration trajectory based on feedforward hidden Markov model [J]. Chinese Journal of Scientific Instrument, 2023,44(12):199-207.

[23] SILVER D, LEVER G, HEESS N, et al. Deterministic policy gradient algorithms[C]. International Conference on Machine Learning, 2014: 387-395.

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

[25] FUJIMOTO S, HOOF H, MEGER D. Addressing

function approximation error in actor-critic methods[C]. International Conference on Machine Learning, 2018: 1587-1596.

[26] HAARNOJA T, ZHOU A, ABBEEL P, et al. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor [C]. International Conference on Machine Learning, 2018: 1861-1870.

作者简介

李逃昌(通信作者),分别在 2006 年和 2009 年于辽宁工程技术大学获得学士学位和硕士学位,2014 年于中国科学院大学获得博士学位,现为辽宁工程技术大学讲师,主要研究方向为智能运动与机器人控制。
E-mail: litaochang@ 163. com

Li Taochang (Corresponding author) received his B. Sc. degree and M. Sc. degree from Liaoning Technical University in 2006 and 2009, and Ph. D. degree from University of Chinese Academy of Sciences in 2014, respectively. Now he is a lecturer in Liaoning Technical University. His main research interests include intelligent motion and robot control.

李健璋,2021 年于辽宁工程技术大学获得学士学位,现为辽宁工程技术大学硕士研究生,主要研究方向为机器人技术与机器学习。
E-mail: lijianzhang0110@ 163. com

Li Jianzhang received his B. Sc. degree from Liaoning Technical University in 2021. Now he is a M. Sc. candidate in Liaoning Technical University. His main research interests include robot technology and machine learning.

侯利民,1999 年和 2004 年分别于辽宁工程技术大学获得学士学位和硕士学位,2010 年于东北大学获得博士学位,现为辽宁工程技术大学教授,主要研究方向为先进控制理论与应用、运动控制。
E-mail: hlm760410@ 163. com

Hou Limin received his B. Sc. degree and M. Sc. degree from Liaoning Technical University in 1999 and 2004, and Ph. D. degree from Northeastern University in 2010, respectively. Now he is a professor in Liaoning Technical University. His main research interests include advanced control theory and its application, motion control.