智能系统学报
CAAI TRANSACTIONS ON INTELLIGENT SYSTEMS

基于深度强化学习的动态装配算法

王竣禾, 姜勇

引用本文:
王竣禾,姜勇. 基于深度强化学习的动态装配算法[J]. 智能系统学报, 2023, 18(1): 2–11.
WANG Junhe,JIANG Yong. Dynamic assembly algorithm based on deep reinforcement learning[J]. *CAAI Transactions on Intelligent Systems*, 2023, 18(1): 2–11.

在线阅读 View online: https://dx.doi.org/10.11992/tis.202201006

DOI: 10.11992/tis.202201006

基于深度强化学习的动态装配算法

王竣禾[1,2,3], 姜勇[1,2]

（1. 中国科学院沈阳自动化研究所 机器人学国家重点实验室, 辽宁 沈阳 110016; 2. 中国科学院机器人与智能制造创新研究院, 辽宁 沈阳 110169; 3. 中国科学院大学, 北京 100049）

摘　要: 针对动态装配环境中存在的复杂、动态的噪声扰动, 提出一种基于深度强化学习的动态装配算法。将一段时间内的接触力作为状态, 通过长短时记忆网络进行运动特征提取; 定义序列贴现因子, 对之前时刻的分奖励进行加权得到当前时刻的奖励值; 模型输出的动作为笛卡尔空间位移, 使用逆运动学调整机器人到达期望位置。与此同时, 提出一种对带有资格迹的时序差分算法改进的神经网络参数更新方法, 可缩短模型训练时间。在实验部分, 首先在圆孔–轴的简单环境中进行预训练, 随后在真实场景下继续训练。实验证明提出的方法可以很好地适应动态装配任务中柔性、动态的装配环境。

关键词: 柔索模型; 动态噪声; 动态装配; 深度强化学习; 长短时记忆网络; 序列贴现因子; 带有资格迹的时序差分算法; 预训练

中图分类号: TP242.6　　**文献标志码:** A　　**文章编号:** 1673−4785(2023)01−0002−10

中文引用格式: 王竣禾, 姜勇. 基于深度强化学习的动态装配算法 [J]. 智能系统学报, 2023, 18(1): 2–11.

英文引用格式: WANG Junhe, JIANG Yong. Dynamic assembly algorithm based on deep reinforcement learning[J]. CAAI transactions on intelligent systems, 2023, 18(1): 2–11.

Dynamic assembly algorithm based on deep reinforcement learning

WANG Junhe[1,2,3], JIANG Yong[1,2]

(1. State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China; 2. Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110169, China; 3. University of Chinese Academy of Sciences, Beijing 100049, China)

Abstract: A dynamic assembly algorithm based on deep reinforcement learning is proposed for complex dynamic noise perturbations in the dynamic assembly environment. Taking the contact force in a period of time as a state, the motion features are extracted through the long short-term memory. Define the sequence discount factor, and obtain the reward value at a certain moment through weighting the sub-reward at the previous moment. The robot can be adjusted to the desired position using inverse kinematics, with the action of model output as the Cartesian space displacement. In the meanwhile, an improved neural network parameter update method is proposed based on the temporal difference (λ) algorithm to shorten the model training time. Experimentally, training was conducted in the real scene upon pre-training in the simple environment with the circular hole-axis. According to the experiments, the proposed algorithm can well adapt to the flexible and dynamic assembly environment in a dynamic assembly task.

Keywords: flexible cable model; dynamic noise; dynamic assembly; deep reinforcement learning; long short-term memory; sequential discount factor; temporal difference(λ); pre-training

机器人控制技术的发展使机器人可以完成多种精密装配任务[1-2]。在这些装配任务中, 作业平台和机器人底座之间的相对位置固定, 机器人在柔顺装配方法的基础上, 根据接触力的不同取值, 做出不同动作。但当作业平台位于柔索之上时, 机器人与作业平台之间会形成紧密的动力学刚柔耦合关系。柔索的位置会因受到机器人的挤压而产生复杂的动态变化, 位置的动态变化又会反过来造成接触力具有瞬时变化的动态噪声, 进而大大影响装配动作的柔顺性。我们将这类装配平台不固定, 且位置受机器人运动影响的装配任务定义为动态装配任务, 柔索装配环境是其中的典型应用场景。如何在动力学刚柔耦合的环境中保证

收稿日期: 2022−01−04.
基金项目: 国家自然科学基金项目(52075531).
通信作者: 姜勇. E-mail: jiangyong@sia.cn.

装配动作的柔顺性, 是动态装配任务中的关键科学问题, 也是工业应用中亟待解决的技术难题。

传统的装配方法从原理上大致可以分为 3 种, 第 1 种是基于物理模型的机器人的装配方法, 这种方法首先通过接触力分析和几何分析得到轴孔三点接触的物理模型, 随后根据模型得到特定接触力下的轴孔位姿偏差, 进而控制机器人进行调整[3]。这种方法对模型建立的精确度有很高的要求, 同时需要环境固定不变。第 2 种方法通过外部机械装置实现了柔顺装配的效果, 麻省理工学院实验室设计了一种远中心柔顺装置[4], 它可以通过内部的弹性元件对环境做出顺应性动作。在此基础上, 根据不同的任务设计不同的刚度自调整策略, 可以使其在装配任务中更具适应性[5]。尽管如此, 在实际使用过程中仍然容易发生卡阻等问题。第 3 种为主动柔顺控制技术, 它从控制理论的角度定义了力与位移的关系, 并且因为可以选择的控制方式多且控制理论发展成熟而得到广泛的应用。Mol 等[6] 提出一种嵌套式的阻抗控制系统来适应不同刚度需求下的装配任务。在大型工件的柔顺装配上, 研究人员使用混合阻抗控制算法, 通过调整阻抗控制的等效接触中心,实现了柔顺装配的效果[7]。主动柔顺控制技术已经可以完成工业场景中大部分装配任务, 但是在面对动态装配环境时, 仍然难以识别因环境动态变化带来的动态噪声。

数据驱动的机器学习方法近年来在机器人操作领域带来了一些显著的成果。相比于传统控制方法, 数据驱动的控制技术可以让机器人从数据中学习。一种基于高斯过程回归的机器人技能学习框架被用到机器人控制中, 使机器人学会了模仿人类动作的技能[8]。广岛大学的团队使用压力传感器的数据进行监督学习, 帮助机器人实现了自主行走[9]。通过数据驱动的方式, 机器人已经能够完成稍微复杂的任务, 伴随着深度强化学习的研究成果从游戏到机器人领域的迁移, 又进一步提高了机器人适应复杂环境的能力。在以往的研究工作中, 机器人不仅实现了抓取物品[10]、开门[11]等一系列复杂任务, 在智能装配领域也涌现了一批重要的研究成果。确定性策略方法被广泛应用于高精度装配任务中[12-13], 取得了令人满意的结果。将传统控制方法与强化学习结合也是实现高精度装配的有效途径[14-15]。在应对环境噪声方面, 通过长短时记忆网络 (long short-term memory, LSTM) 网络提取动态信息这一技巧被广泛应用到机器人领域[16-18], Inoue 等[19] 使用 LSTM 网络进行特征提取, 并利用深度 Q 网络训练神经网络控制器, 提高了任务的成功率。将人类专家数据引入强化学习模型也可以增加控制的鲁棒性[20]。

在以往的装配研究中, 环境的动态性主要来源于轴孔之间的摩擦力、机器人关节噪声、机器人定位误差、传感器固有噪声等, 这种情况下, 我们会得到带有噪声的接触力值 。虽然噪声来源众多, 但是由于装配平台都是固定的, 因此噪声值都很小。但在刚柔动力学紧密耦合的动态装配任务中, 柔索环境高柔性和高动态的特点, 使装配平台处于一种动态变化的状态, 也使得接触力噪声具有瞬时且动态变化的特征。鉴于以上这些因素, 设计了此基于深度强化学习的动态装配控制算法。在模型设计阶段, 为提高模型抗干扰能力, 将一段时间内的接触力作为状态值, 并利用 LSTM 网络对其进行特征提取; 采用基于运动趋势的分奖励设计, 在序列折扣因子α的加权下得到每一时刻的奖励; 使用能够处理连续动作的 Actor-Critic 算法进行训练; 借鉴带有资格迹的时序差分算法 (temporal difference(λ), TD(λ)) 算法参数更新方法, 并对其提出改进, 加快了训练速度。

输电线路上的绝缘子串在经受长期的风吹日晒后容易出现锈蚀、龟裂等问题[21-22], 这会对输电线路的安全造成很大的威胁, 需要定期进行检修和更换。由于输电线路具有悬链线特征, 属于典型的柔索模型[23-24], 因此对机器人更换绝缘子串任务中的动态装配问题进行分析, 并通过实验检验算法的有效性。

在仿真实验阶段, 通过对工件位置增加噪声来模拟输电线路和机器人实际接触过程中发生的动态变化。同时为了使控制器学习到核心技能, 忽略建模差异带来的影响, 使用圆孔、轴的装配环境进行预训练。待训练完成后, 将其迁移到真实的装配环境中再次训练, 直到模型收敛。

1　动力学刚柔耦合分析

如图 1 所示, 输电线路上的两片绝缘子之间通过绝缘销固定, 实现绝缘子更换的第一步是将销孔中的销子推出。完成推销任务的机器人位于绝缘斗臂车的作业平台上, 其末端连杆上装有推销工具。在装配开始前, 首先操作机器人使推销工具在视觉的引导下粗略得到达销孔上方。当多刚体机器人与位于柔性输电线路上的销孔发生接触时, 输电线路的位置会因受到挤压发生动态变化, 进而改变销孔与机器人底座的相对位置关系, 以此引发的动态冲击力将导致接触力动态变化。如果此时片面地增大机器人动作的调整量, 可能使输电线路位置发生震荡, 以致最终完全失去控制。本文提出的方法通过分析一段时间内接

触力的变化趋势，使机器人在保持装配柔顺性的同时逐步吸收刚柔动力学耦合系统中的作用力。

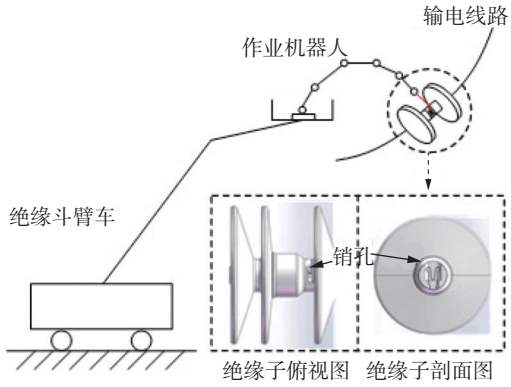


图 1　装配环境示意图

Fig. 1　Schematic diagram of assembly environment

2　动态装配算法设计

2.1　马尔可夫决策过程

利用强化学习方法解决此类问题，需要构建包括状态、动作、奖励在内的马尔可夫决策过程(图 2)。

在状态设计方面，为了识别动态噪声并减小其对控制系统的影响，将$[t-k+1,t]$时间范围内共k个接触力值组成的力序列定义为状态，并使用 LSTM 提取其中主要的运动趋势信息。图 3 给出了装配过程中，状态中某一维度随时间的变化过程。

将状态定义为$\boldsymbol{s}$，则t时刻的状态向量$\boldsymbol{s}_t=[\boldsymbol{f}_{t-k+1}\ \boldsymbol{f}_{t-k+2}\ \cdots\ \boldsymbol{f}_t]^{\mathrm{T}}$，其中$\boldsymbol{f}_t=[f_x\ f_y\ f_z]^{\mathrm{T}}$。图 4 给出了在没有使用装配算法的时，轴孔之间不同运动趋势对应的状态表示。其中轴的初始化位置为销孔中心。黄色采样点表示轴孔之间无相对运动，蓝色采样点显示了轴孔相互接触的运动过程，红色采样点显示了轴孔由接触到分离的运动过程。可以看出，虽然噪声的存在可能会对局部范围内接触力的取值造成影响，但接触力的整体变化趋势与运动趋势相同。同时，此图证明了不同状态在状态空间具有明显不同的特征，这种设置也为提取运动特征的 LSTM 网络部分提供了具有区分度的样本。

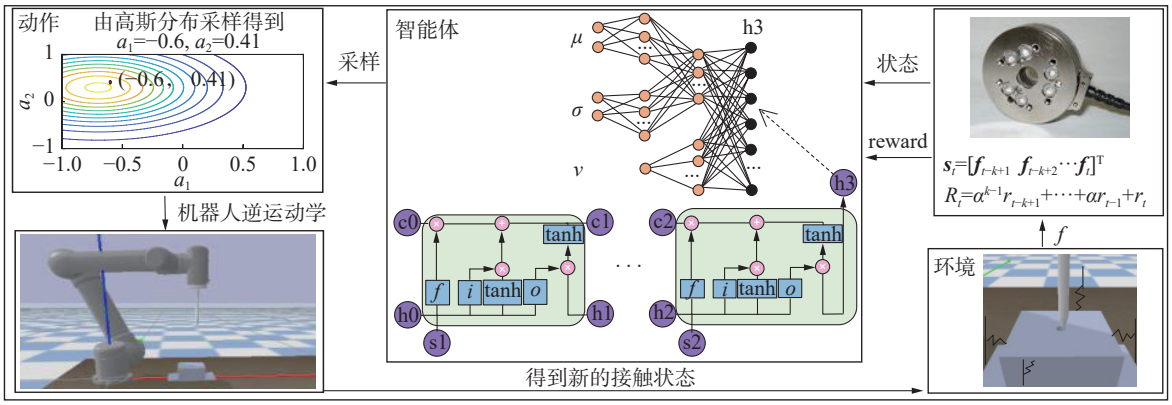


图 2　马尔可夫决策过程

Fig. 2　Markov decision process

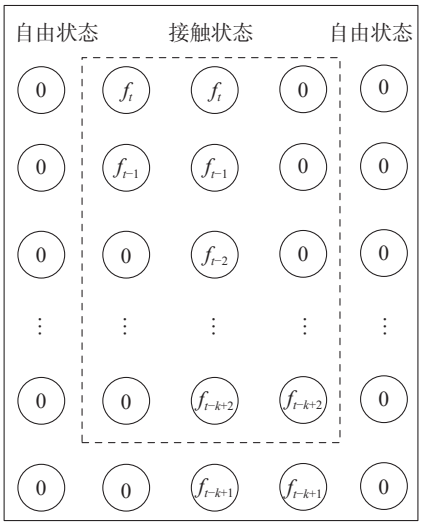
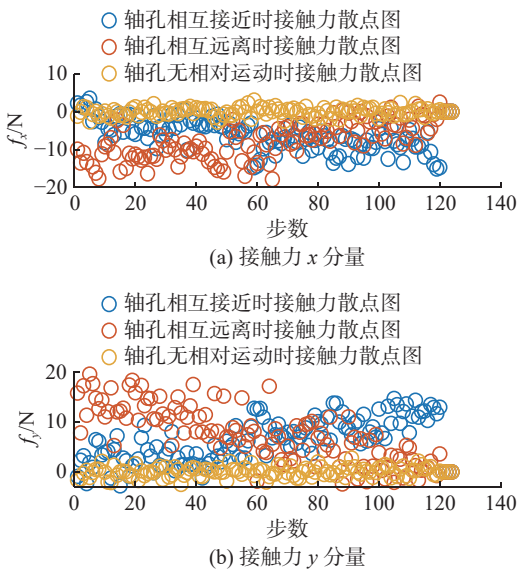
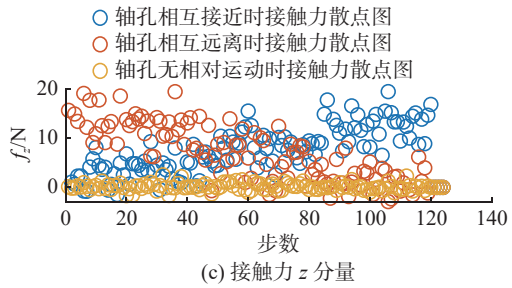


图 3　状态中一维特征随时间变化情况

Fig. 3　One dimensional feature in state change with time





(c) 接触力 z 分量

图 4　不同运动趋势时接触力各个分量变化情况

Fig. 4　Different components of contact force in different motion trends

将机器人下一时刻沿着末端坐标系x轴和y轴方向移动的距离定义为动作, 用二维向量$\boldsymbol{a}$进行表示, 在t时刻$\boldsymbol{a}_t=[a_1\ a_2]$。

在奖励函数的设计中, 首先基于运动趋势定义每一步的分奖励r, 然后使用序列折扣因子α对这之前k步中每一步的分奖励进行加权, 得到当前时刻的奖励R。在分奖励的设计中, 首先根据当前接触力绝对值的最大值是否大于阈值f_{contact}判断轴孔是否接触。当轴孔接触时, 分奖励设置为前后两次接触力中, 3 个维度绝对值差之和的相反数, 用 sum 表示求和; 当轴孔处于非接触状态时, 分奖励大小为一固定值 c。

$$r_t=\begin{cases}-\text{sum}(\Delta|\boldsymbol{f}|), & \max(|\boldsymbol{f}|)\geqslant f_{\text{contact}}\\ c, & \max(|\boldsymbol{f}|)<f_{\text{contact}}\end{cases}\tag{1}$$

由于状态是由k个时刻的接触力组成, 对应的奖励也是由此前k个时刻的分奖励构成。对分奖励按照其距离当前时刻的远近, 赋予不同权重。将这些带权重的分奖励加和就得到了每一时刻的奖励R_t。

$$R_t=r_t+\alpha r_{t-1}+\cdots+\alpha^{k-1}r_{t-k+1}\tag{2}$$

完成此次任务机器人是末端搭载六维力传感器的 UR5 机械臂。六维力传感器采集接触力信息并组成状态$\boldsymbol{s}$, 将其输入到深度强化学习模型(agent)后, agent 输出动作的均值和方差, 采样后得到动作$\boldsymbol{a}$, 机器人经过逆运动学计算后得到期望关节角度, 控制电机运动到指定角度后, 环境会根据轴孔接触情况给出相应的奖励R和下一时刻的状态$\boldsymbol{s}'$, 随后 agent 开始更新参数, 并开始下一轮的决策控制。

2.2　基于深度强化学习装配算法设计

2.2.1　Actor-Critic 算法

将 agent 在状态$\boldsymbol{s}$处基于策略π得到的累积期望奖励表示为J, 其中的策略π为使用w_1参数化的神经网络表示的高斯分布, Q 为状态动作价值函数。

$$J_t=\text{E}_\pi\left[\sum\nolimits_a Q(\boldsymbol{s}_t,\boldsymbol{a})\pi(\boldsymbol{a}|\boldsymbol{s}_t;w_{1t-1})\right]\tag{3}$$

通过梯度上升的方法优化参数w_1, 可以使 agent 获得最大的累计期望奖励。

$$\nabla J_t=\text{E}_\pi\left[\sum\nolimits_a Q(\boldsymbol{s}_t,\boldsymbol{a})\nabla\pi(\boldsymbol{a}|\boldsymbol{s}_t;w_{1t-1})\right]\tag{4}$$

使用蒙特卡洛采样代替对动作求期望, 并为了减小方差加入基础项b, 可以将式 (4) 改写为

$$\nabla J_t=\text{E}_\pi[(G_t-b)\nabla\ln\pi(\boldsymbol{a}_t|\boldsymbol{s}_t;w_{1t-1})]\tag{5}$$

将 TD 的更新方式引入式 (5) 后, 可以实现在线更新。同时使用w_2参数化的值函数网络表示基础项b, 也令梯度设计更加合理。于是G_t-b便可以用 TD 误差$\delta_{t,w_{2t-1}}$代替, 其中γ表示折扣因子。

$$\delta_{t,w_{2t-1}}=Q(\boldsymbol{s}_t,\boldsymbol{a}_t)-v(\boldsymbol{s}_t)=R_t+\gamma v(\boldsymbol{s}_{t+1};w_{2t-1})-v(\boldsymbol{s}_t;w_{2t-1})$$

将$\delta_{t,w_{2t-1}}$代入式 (5), 得到 Actor-Critic 算法下的策略梯度更新公式:

$$\nabla J_t=\text{E}_\pi[\delta_{t,w_{2t-1}}*\nabla\ln\pi(\boldsymbol{a}_t|\boldsymbol{s}_t;w_{1t-1})]$$

值函数损失定义为均方误差形式, 值函数网络的输出直接决定了δ的取值, 会对策略更新起到重要影响, 因此为了使值函数网络的参数尽快收敛到正确值, 在值函数更新阶段借鉴了TD(λ)算法, 通过对其改进, 进一步加快收敛速度。

2.2.2　值函数网络参数更新

TD(λ)算法以时序差分的方式, 通过资格迹向量将当前 TD 误差以不同的贡献分配给之前状态对应的值函数梯度, 来提高更新的稳定性[25]。首先定义资格迹向量$\boldsymbol{z}$以及更新方向$\boldsymbol{p}$, 然后可以得到它们在t时刻的取值。其中λ表示资格迹因子。

$$\boldsymbol{z}_t=\nabla v(\boldsymbol{s}_t,w_{2t-1})+\lambda\gamma\boldsymbol{z}_{t-1}$$

$$\boldsymbol{p}_t=\boldsymbol{z}_t*\delta_{t,w_{2t-1}}$$

将时刻状态值函数梯度对更新方向的总贡献定义为$\boldsymbol{P}_m$, 来分析到时刻为止, m 时刻状态值函数的梯度后续更新的影响, 其中$t>m$。从下面公式可以发现, 由于γ和λ均小于 0, 因此随着时间的推移, 时刻对应的状态值函数梯度所分配到的TD(λ) 误差也越来越小。这意味着它和之后时刻的状态梯度之间的相似度也越来越小。在这种权值分配方式的基础上, 对时刻价值函数梯度的计算方式提出改进, 使本应由时刻的参数计算的价值函数梯度变为使用实时更新的参数计算。

$$\boldsymbol{P}_m=\nabla v(\boldsymbol{s}_m,w_{2m-1})\delta_{m,w_{2m-1}}+\gamma\lambda\nabla v(\boldsymbol{s}_m,w_{2m-1})\delta_{m+1,w_{2m}}+\cdots+\gamma^{t-1}\lambda^{t-1}\nabla v(\boldsymbol{s}_m,w_{2m-1})\delta_{t,w_{2t-1}}\tag{6}$$

因为距离当前时刻越近的状态, 其值函数输出也就越接近, 当使用相同参数时, 其梯度在特征空间的相似度差异就越小。同时由于使用实时更新的参数计算过去时刻状态值函数的梯度, 也可以提高样本利用率并加快更新速度。具体实现时, 额外使用长度为 L_{zige} 的状态队列$\boldsymbol{s}_{\text{queue}}$和权重队列$\boldsymbol{w}_{\text{queue}}$, 分别保存每个时刻的状态和对应的权

重系数，具体参数更新伪代码如下所示。

Init $\boldsymbol{s}$, $\boldsymbol{a}$, γ, λ, $\boldsymbol{s}_{\text{queue}}$, $\boldsymbol{w}_{\text{queue}}$, w_2

Init ZIGE_LEN, LEARNING_RATE, max_num, env, net

Repeat: max_num

  while(!done)

    $\boldsymbol{s}'$, R, done=env($\boldsymbol{a}$)

    if len($\boldsymbol{s}_{\text{queue}}$)< ZIGE_LEN

      $\boldsymbol{s}_{\text{queue}}$.append($\boldsymbol{s}$)

      $\boldsymbol{w}_{\text{queue}}$.append($\gamma\lambda^{(\text{len}(\boldsymbol{s}_{\text{queue}})-1)}$)

    else

      $\boldsymbol{s}_{\text{queue}}$.pop(0)

      $\boldsymbol{s}_{\text{queue}}$.append($\boldsymbol{s}$)

    $\delta = R + \gamma v(\boldsymbol{s}'; w_{2t-1}) - v(\boldsymbol{s}; w_{2t-1})$

    $\boldsymbol{g} = \text{reverse}(\boldsymbol{w}_{\text{queue}}) * \nabla net(\boldsymbol{s}_{\text{queue}})$

    $\boldsymbol{p} = \text{LEARNING\_RATE} * \boldsymbol{g} * \delta$

图 5 给出了值函数网络的更新流程。其中上面虚线框中的流程为模型根据环境状态给出动作，进而使仿真得以一直运行；在下面的虚线框中，首先将当前状态压入状态队列中，然后由模型得到当前状态梯度，再在权重队列的作用下得到此次的最终梯度 $\boldsymbol{g}_t$，最终得到更新后的参数。

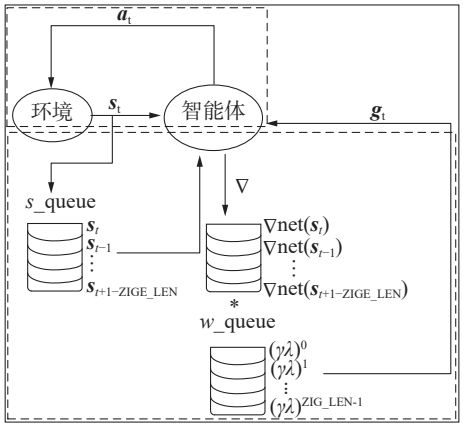


图 5　值函数网络更新流程图

Fig. 5　Value function network update flow chart

3　实验验证

3.1　仿真系统设置

3.1.1　仿真环境

Bullet 引擎提供了许多机器人控制接口以及灵活模型导入方式，因此采用 Bullet 引擎搭建仿真环境。其中轴孔模型由 SolidWorks 绘图软件绘制。由于本实验对装配精度要求不高，因此设置圆孔的半径为 0.006 3 m，轴体半径为 0.005 8 m，两者间隙为 0.001 m。使用 Bullet 引擎自带末端六维力传感器采集接触力数据，在实验开始前，需要设置传感器坐标系方向与机器人末端坐标系方向相同，同时通过 Bullet 提供的函数接口将重力加速度设置为 0 m/s^2，以此来达到当轴孔不接触时，力传感器读数为 0 N 的目的。算法实现的软件部分由 Pytorch 实现，在 NVIDIA GTX1050ti 上完成训练。

3.1.2　初始位置设置

为了使 agent 得到充分的训练，避免陷入局部极值，在每一幕开始时，需要随机初始化轴的落点 (图 6)。由于待装配的孔为圆形，因此这里采用两步走的随机化方式：首先从零均值的高斯分布中采样，得到落点离孔圆心的距离 d；然后从 $U[0,d]$的均匀分布中采样得到落点的 x 坐标，最后由余弦定理得到 y 坐标。

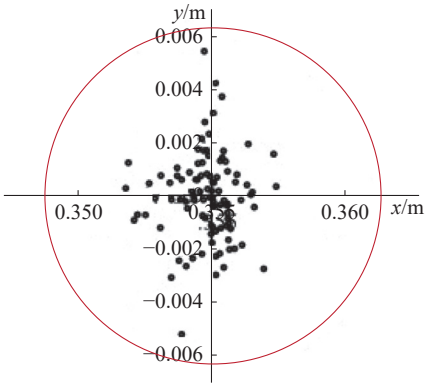


图 6　随机初始化落点位置图

Fig. 6　Location map of random initial landing point

由图 2 可知，装配环境为动态的，可以近似为与弹簧固连的整体，在本实验中，以相等的概率在孔中心坐标的 x 轴分量或 y 轴分量上添加随机噪声，并随机化噪声出现的时间，设置噪声出现的时间服从均匀分布 $U[0,6]$，大小服从零均值的高斯分布。

3.2　结果

3.2.1　模型训练

输入到模型中的样本大小为 $k\times3$，其中 k 为组成状态的力序列长度。采用 3 个 LSTM 单元提取轴孔之间的运动趋势。每个 LSTM 单元输入为长度为 k 的特征向量，用于提取运动过程中一个维度的接触力变化情况。在网络内部采用双隐层设计，每层有 20 个神经元，并使用 dropout 技术防止模型过拟合，每次反向传播时有 30% 的神经元被抑制。数据经过 LSTM 处理后被传送到全连接神经网络中，进而输出由动作均值向量和动作方差向量组成的策略函数，以及状态价值函数。总的损失函数由状态值函数相关的均方误差损失以及策略函数相关的策略梯度损失和熵损失 3 部分组成。具体的深度神经网络数学模型为

$$\text{out1} = \text{LSTM}(k,20,20,0)$$
$$\text{out2} = \text{LSTM}(k,20,20,\text{out1})$$
$$\text{out3} = \text{LSTM}(k,20,20,\text{out2})$$
$$\text{out4} = \text{MLP}(\text{out3},10), v = \text{MPL}(\text{out4},1)$$
$$\text{out5} = \text{MLP}(\text{out3},10)$$
$$\mu = \text{MPL}(\text{out5},2), \sigma = \text{MLP}(\text{out5},2)$$

式中: LSTM 层激活函数采用经典的 sigmod 函数和 tanh 函数方式配置, 全连接 MLP 层的隐含层 out4 和 out5 采用 Relu 激活函数, v 采用 sigmod 激活函数, μ 采用 PRelu 激活函数, σ 采用 Softplus 激活函数。数学模型对应网络配置见图 2。

图 7、图 8 给出了在训练过程中, k 分别 13、15 和 17 时, 平均步数和平均奖励随回合数增多的变化曲线, 从中可以看到, 当 k 取 15 时, 模型取得较高的回合平均步数和回合平均奖励, 意味着网络可以能够较好得提取运动特征, 做出符合当前状态的动作。

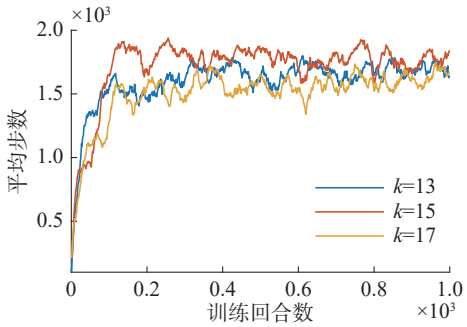


图 7　平均步数随训练变化曲线

Fig. 7　Variation curve of average steps with training

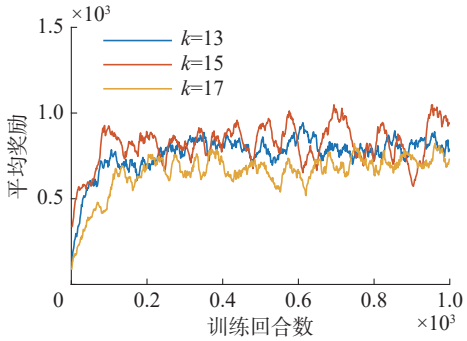


图 8　平均奖励随训练变化曲线

Fig. 8　Variation curve of average rewards with training

分别对比了在资格迹长度为 100, 资格迹因子为 0.75 的情况下, 值函数参数更新方法和传统 TD(λ)算法在训练时间上的差异。由图 9 可以看到, 使用本文方法计算的损失由初始值收敛到 0 附近的时间比传统的TD(λ)算法短。从代码执行的角度, 是一种用空间换时间的思想。当状态空间不是特别大时, 是可以实现高效运行的。随着训练的继续, 值函数网络的参数已经收敛, 此时两者的损失也相继收敛到 0 附近。

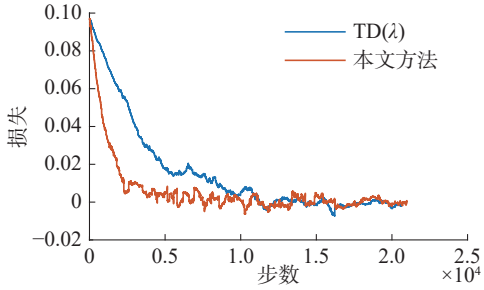


图 9　不同参数更新方式曲线

Fig. 9　Curves with different parameter updating methods

在模型训练过程中, 以固定的回合间隔对模型性能进行测试, 由于在模型训练初期参数的鲁棒性较差, 容易受到环境噪声的干扰而陷入局部极值, 进而造成一段时间内测试性能一直没有提升。因此当测试时间间隔大于阈值时, 调用在此之前的保存的次优模型参数重新进行训练, 从而及时纠正模型更新方向, 减小模型训练时间, 最终总体训练时间为 5 h。

训练过程中用到的关键参数有折扣因子γ, 序列折扣因子α, 资格迹因子λ, 批量元素个数 S_{batch}, 学习率 R_{learning}, 测试间隔 I_{test}, 资格迹向量长度 L_{zige}, 构成状态的力序列长度 L_{seq} 以及导入次优参数的等待时间 T_{wait}。参数的具体取值见表 1。

表 1　模型关键参数设置

Table 1　Model key parameter setting

参数名称	数值
γ	0.97
α	0.95
λ	0.75
S_{batch}	12
R_{learning}	3e-5
I_{test}	4
L_{zige}	100
L_{seq}	15
T_{wait}	300

3.2.2　模型测试

孔中心点坐标为 pos_origi(0.355 m, 0 m), 在以孔中心点为圆心的同心圆上等间距取 4 个点作为测试时的落点位置, 分别定义为 pos1(0.357 m, 0 m), pos2(0.355 m, 0.002 m) , pos3(0.353 m, 0 m), pos4 (0.355 m, −0.02 m) 。每次测试需要经过 4 个回合, 这 4 个回合分别以这 4 个点作为机器人末端坐标系的初始位置。分别计算每个回合的步长和奖励, 得到平均步长和平均奖励(图 10、图 11)。当某次测试得到的平均步长大于之前最好的结果时, 便可以将当时的模型参数保存。训练完成

后，将机器人末端坐标系位置移动到 pos1 点，使用最优参数对应的模型，分别在具有环境噪声和不具有环境噪声的情况下进行装配实验，得到x轴的力反馈散点图及模型输出动作散点图。

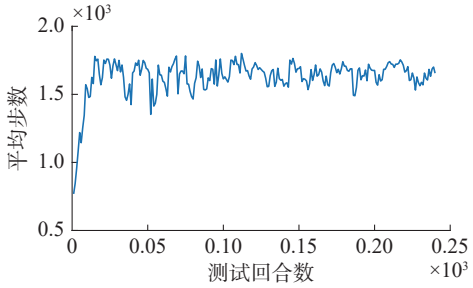


图 10　平均步数随测试变化曲线

Fig. 10　Variation curve of average steps with testing .

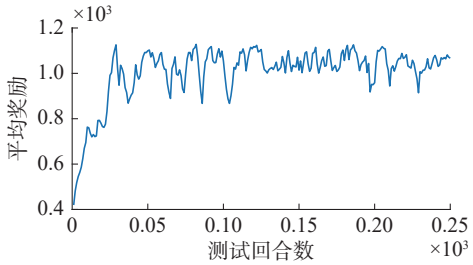


图 11　平均奖励随测试变化曲线

Fig. 11　Variation curve of average rewards with testing

当没有动态噪声时，如图 12、13 所示，一开始接触力为 0，表示轴孔并无接触，此时 agent 输出的动作也近似为 0；随后轴孔开始接触，接触力由 0 开始逐渐增大，当 agent 能够根据状态向量中接触力的变化情况提取到主要的运动信息后，将输出动作。机器人执行动作后，会重新回到轴孔分离的状态，此时接触力又重新归 0，输出的动作大小也为 0。

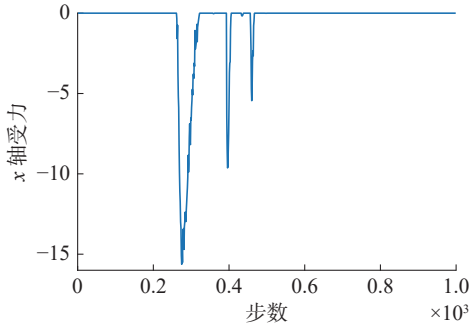


图 12　无动态噪声时 x 轴受力曲线

Fig. 12　*X*-axis force curve without dynamic noise

当接触力中存在由动态冲击力引发的动态噪声时，如图 14、15 所示，可能会导致轴孔之间的接触力出现突然增加的情况，但是由于噪声的存在时间短，因此相比于正常装配输出的动作幅值，此时 agent 并不会输出太大的动作。由此本文认为，agent 学习到的这种机制可以很好地应对柔性动态环境，能够保证装配的安全性。图 16 展示了仿真环境中装配过程片段。

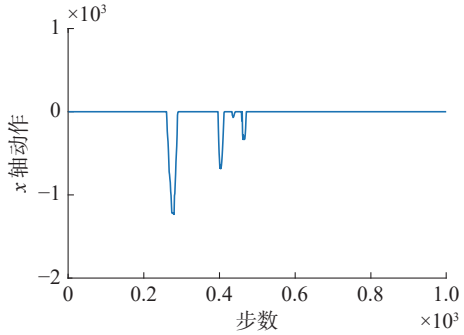


图 13　无动态噪声时 x 轴动作曲线

Fig. 13　*X*-axis action curve without dynamic noise

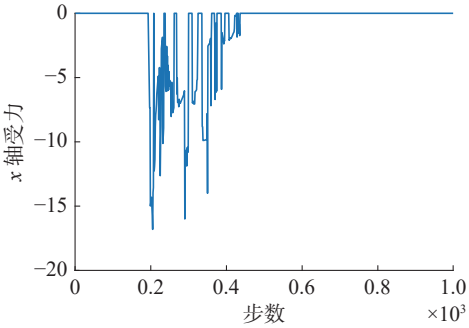


图 14　有动态噪声时 x 轴受力曲线

Fig. 14　*X*-axis force curve with dynamic noise

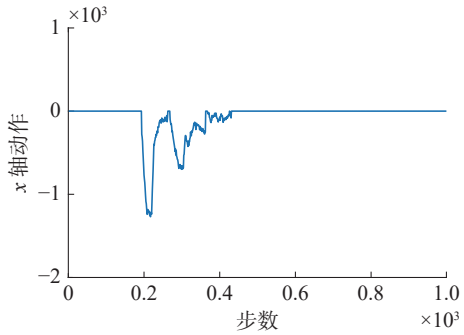
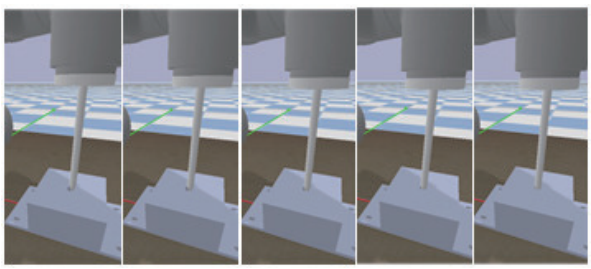


图 15　有动态噪声时 x 轴动作曲线

Fig. 15　*X*-axis action curve with dynamic noise



(a) 片段 1　(b) 片段 2　(c) 片段 3　(d) 片段 4　(e) 片段 5

图 16　仿真实验装配片段

Fig. 16　Simulation experiment assembly segment

3.3　真实实验设置

3.3.1　实验环境

如图 17 所示, 使用 NVIDIA 公司的 Jetson Xavier 作为主控单元, 使用 UR5 机械臂进行此次装配实验, 通过 RealsenseD435i 相机进行装配前的路径规划。机器人末端搭载 ATI 公司的 mini45 六维力传感器进行数据采集。在软件设计部分, 使用 KDL 机器人运动学库和 UR_Script 语言进行机器人逆运动学求解以及对机械臂进行控制。

图 17　实验平台图
Fig. 17　Experimental platform diagram

3.3.2　实验分析

在仿真实验中已经使用深度强化学习方法, 解决了圆孔环境下的动态装配问题。在此基础上将在仿真中获得的最优参数对应的模型看作是真实环境的预训练模型, 然后利用这个预训练模型在现实装配环境中再次训练, 最终使 agent 在掌握核心策略的基础之上更加适合真实的作业环境。在实验开始前, 首先需要对传感器进行参数辨识, 以得到轴重力、传感器温漂等力偏差值, 以及传感器与机器人末端坐标系之间的偏角。然后将辨识得到的力偏差补偿到从传感器直接获取的力信息中, 确保状态中的力仅为接触力。虽然预训练模型有良好的参数基础, 但是由于真实场景中信息传输的整体速率比仿真中低, 且 jetson 运算速度低于独立显卡, 因此模型收敛总耗时大约 3 h 实验过程如图 18 所示。

图 18　真实实验片段
Fig. 18　Real experimental fragment

在使用传统的柔顺控制方法解决此类问题时, 首先需要构建二阶柔顺系统模型, 通过求解微分方程的方式, 可以由接触力得到当前的位移增量, 由于柔顺控制系统参数固定不变, 不能识别动态的装配场景, 因此会对突然增大的接触力有强烈的反应。因此本文分别使用传统柔顺控制方法和本文提到的方法进行装配实验, 得到各自的接触力曲线 (图 19、20)。

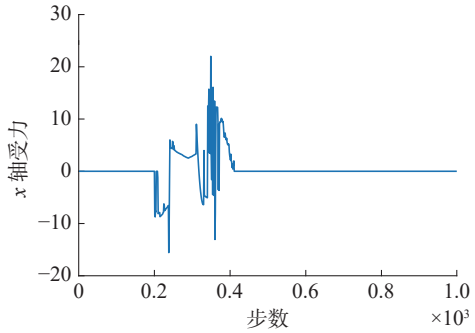


图 19　使用柔顺控制方法装配得到的 *x* 轴受力曲线
Fig. 19　*X*-axis force curve by using compliance control

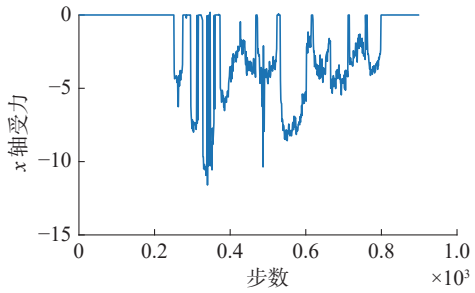


图 20　使用本文提出方法装配得到的 *x* 轴受力曲线
Fig. 20　*X*-axis force curve by using ours

由图 20 中可以看出, 使用本文方法得到的接触力方向始终不变, 即 agent 输出的动作变化不大; 反观图 19 中使用传统柔顺控制方法得到的接触力曲线, 其值在正负之间不断变化, 说明动态噪声的存在使系统出现震荡。虽然其调节时间比本文方法短, 但是由于极其不稳定, 因此不但不能保证装配成功率(本文方法成功率为 0.82, 柔顺控制方法成功率为 0.52), 还易造成工件损坏。图 21 给出了将本文方法应用到输电线路绝缘子更换任务中的场景。实验证明所提方法具有很好的稳定性和实用性。

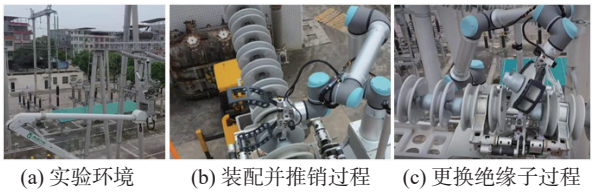


图 21　输电线路绝缘子更换实验
Fig. 21　Experiment of insulator replacement on transmission line

4　结束语

本文关注动态装配任务中尚未得到很好的解决的问题，即：作业平台高柔性、高动态的特点，使得传统方法在解决动态装配任务时很容易发生震荡。

对此本文提出一种基于深度强化学习的动态装配控制算法。首先构建了马尔可夫决策过程，基于接触力噪声虽然响应值大，但是一次持续时间短的特征，将之前k个时刻接触力组成的力序列作为状态，并使用带有 LSTM 单元的网络提取过去一段时间内接触力的主要变化趋势，最终输出代表x和y方向位移的动作。在奖励函数的设计上，首先计算每一步的分奖励，然后使用序列折扣因子对这之前k步的分奖励进行加权得到当前时刻奖励。使用 Actor-Critic 算法对 agent 进行训练，通过使用额外的存储空间的方法，更改TD(λ)算法中过去状态的梯度的计算方式，当状态空间维度不是特别高时，可以加快了训练速度。实验部分通过预训练模型进行核心技能学习，然后放在真实环境中再次训练以适应具体环境。最后通过输电线路绝缘子的更换任务，检验了提出的方法可以有效地处理动态环境下的装配问题，能够在解决动态噪声引起的震荡的同时，以较高的成功率完成动态装配任务。

参考文献：

[1] LI Fengming. Robot skill acquisition in assembly process using deep reinforcement learning[J]. Neurocomputing, 2019, 345: 92–102.

[2] WANG Zichen, YANG Xiansheng, HU Haopeng, et al. Actor-critic method-based search strategy for high precision peg-in-hole tasks[C]//2019 IEEE International Conference on Real-time Computing and Robotics. Irkutsk: IEEE, 2019: 458−463.

[3] TE Tang, LIN H C, ZHAO Yu, et al. Autonomous alignment of peg and hole by force/torque measurement for robotic assembly[C]//2016 IEEE International Conference on Automation Science and Engineering. Fort Worth: IEEE, 2016: 162−167.

[4] ROURKE J M, WHITNEY D E. Remote center compliance device: US4556203[P]. 1985−12−03.

[5] LEE S. Development of a new variable remote center compliance (VRCC) with modified elastomer shear pad (ESP) for robot assembly[J]. IEEE transactions on automation science and engineering, 2005, 2(2): 193–197.

[6] MOL N, SMISEK J, BABUŠKA R, et al. Nested compliant admittance control for robotic mechanical assembly of misaligned and tightly toleranced parts[C]//2016 IEEE International Conference on Systems, Man, and Cybernetics. Budapest: IEEE, 2016: 2717−2722.

[7] HE Gang, SHI Shicai, WANG Da, et al. A strategy for large workpiece assembly based on hybrid impedance control[C]//2019 IEEE International Conference on Mechatronics and Automation. Tianjin: IEEE, 2019: 799−804.

[8] FORTE D, UDE A, KOS A. Robot learning by Gaussian process regression[C]//19th International Workshop on Robotics in Alpe-Adria-Danube Region. Budapest: IEEE, 2010: 303−308.

[9] BHATTACHARYA S, DUTTA S, MAITI T K, et al. Machine learning algorithm for autonomous control of walking robot[C]//2018 International Symposium on Devices, Circuits and Systems. Howrah: IEEE, 2018: 1−4.

[10] FINN C, LEVINE S. Deep visual foresight for planning robot motion[C]//2017 IEEE International Conference on Robotics and Automation. Singapore: IEEE, 2017: 2786−2793.

[11] NEMEC B, ŽLAJPAH L, UDE A. Door opening by joining reinforcement learning and intelligent control[C]//2017 18th International Conference on Advanced Robotics. Hong Kong: IEEE, 2017: 222−228.

[12] XU Jing, HOU Zhimin, WANG Wei, et al. Feedback deep deterministic policy gradient with fuzzy reward for robotic multiple peg-in-hole assembly tasks[J]. IEEE transactions on industrial informatics, 2019, 15(3): 1658–1667.

[13] HE Fujun, WANG Xiaozheng, LIU Kai. Research on axle-hole assembly method based on improved DDPG algorithm[C]//2021 5th International Conference on Robotics and Automation Sciences. Wuhan: IEEE, 2021: 182−186.

[14] ROVEDA L, PALLUCCA G, PEDROCCHI N, et al. Iterative learning procedure with reinforcement for high-accuracy force tracking in robotized tasks[J]. IEEE transactions on industrial informatics, 2018, 14(4): 1753–1763.

[15] ZHOU Zhenning, NI Peiyuan, ZHU Xiaoxiao, et al. Compliant robotic assembly based on deep reinforcement learning[C]//2021 International Conference on Machine Learning and Intelligent Systems Engineering. Chongqing: IEEE, 2021: 6−9.

[16] NAGURI C R, BUNESCU R C. Recognition of dynamic hand gestures from 3D motion data using LSTM and CNN architectures[C]//2017 16th IEEE International Conference on Machine Learning and Applications. Cancun: IEEE, 2017: 1130−1133.

[17] WU Zhixuan, MA Nan, CHEUNG Y M, et al. Improved spatio-temporal convolutional neural networks for traffic police gestures recognition[C]//2020 16th International Conference on Computational Intelligence and Security. Guangxi: IEEE, 2020: 109−115.

[18] ZHANG Weihu, LIU Chang. Research on human abnormal behavior detection based on deep learning[C]//2020 International Conference on Virtual Reality and Intelligent Systems. Zhangjiajie: IEEE, 2020: 973−978.

[19] INOUE T, DE MAGISTRIS G, MUNAWAR A, et al. Deep reinforcement learning for high precision assembly tasks[C]//2017 IEEE/RSJ International Conference on Intelligent Robots and Systems. Vancouver: IEEE, 2017: 819−825.

[20] MA Yanqin, XU De, QIN Fangbo. Efficient insertion control for precision assembly based on demonstration learning and reinforcement learning[J]. IEEE transactions on industrial informatics, 2021, 17(7): 4492–4502.

[21] SUTTON Richard Stuart. Temporal credit assignment in rein-forcement learning[D]. Amherst: University of Massachusetts Amherst, 1984: 93−118.

[22] 车立新, 杨汝清, 顾毅. 220/330kV 变电设备高压带电清扫机器人设计 [J]. 机器人, 2005, 27(2): 102–107.
CHE Lixin, YANG Ruqing, GU Yi. Design of high-voltage hot-line sweeping robot used in 220/330kV substation[J]. Robot, 2005, 27(2): 102–107.

[23] 周松. 高压输电线内力及变形分析的有限元法 [J]. 四川电力技术, 1995, 18(2): 6–10.
ZHOU Song. Finite element method for analysis of internal force and deformation of High voltage transmission lines[J]. Sichuan electric power technology, 1995, 18(2): 6–10.

[24] 魏永乐, 房立金. 双臂巡检机器人沿输电线路行走特性研究 [J]. 北京理工大学学报, 2019, 39(8): 813–818.
WEI Yongle, FANG Lijin. Research on dual-arms inspection robots walking along transmission line[J]. Transactions of Beijing Institute of Technology, 2019, 39(8): 813–818.

[25] MATEUS C, BARATA F A, LUÍS R. Effects of broken skirts and pollution on voltage distribution for cap and pin glass insulators[C]//2020 IEEE 14th International Conference on Compatibility, Power Electronics and Power Engineering. Setubal: IEEE, 2020: 30−35.

作者简介:

王竣禾, 硕士研究生, 主要研究方向为强化学习、智能机器人。

姜勇, 研究员, 主要研究方向为机器人智能控制、适于复杂环境的机器人遥操作、嵌入式控制系统与应用、多传感器融合与系统健康管理、人机协同控制理论与方法、特种机器人控制系统设计与集成。负责及参加完成了国家 863 重点项目、国家自然科学基金青年及面上项目、中科院知识创新工程重大项目、辽宁省自然科学基金项目、机器人学重点实验室项目、国网及南网重点项目等 20 多项, 申请国家发明专利 3 项, 实用新型专利 4 项, 登记软件著作权 2 项。参加编写专著 2 部, 发表学术论文 20 多篇。