DOI:10.19651/j.cnki.emt.2315190

基于强化学习的多段连续体机器人轨迹规划*

刘宜成[1] 杨迦凌[1] 梁 斌[2] 陈 章[2]
(1.四川大学电气工程学院 成都 610065; 2.清华大学自动化系 北京 100084)

摘 要: 针对多段连续体机器人的轨迹规划问题,提出了一种基于深度确定性策略梯度强化学习的轨迹规划算法。首先,基于分段常曲率假设方法,建立连续体机器人的关节角速度和末端位姿的正向运动学模型。然后,采用强化学习算法,将机械臂的当前位姿和目标位姿等信息作为状态输入,将机械臂的关节角速度作为智能体的输出动作,设置合理的奖励函数,引导机器人从初始位姿向目标位姿移动。最后,在 MATLAB 中搭建仿真系统,仿真结果表明,强化学习算法成功对多段连续体机器人进行轨迹规划,控制连续体机器人的末端平稳运动到目标位姿。

关键词: 连续体机器人;轨迹规划;强化学习;位姿控制;奖励引导
中图分类号: TP242;TP399 **文献标识码**: A **国家标准学科分类代码**: 520.60

Trajectory planning of multi-stage continuum robot based on reinforcement learning

Liu Yicheng[1] Yang Jialing[1] Liang Bin[2] Chen Zhang[2]
(1. College of Electrical Engineering, Sichuan University, Chengdu 610065, China;
2. Department of Automation, Tsinghua University, Beijing 100084, China)

Abstract: For the trajectory planning of multi-stage continuum robots, a trajectory planning algorithm based on deep deterministic policy gradient reinforcement learning is proposed. Firstly, based on the piecewise constant curvature hypothesis, the forward velocity kinematic model of joint angular velocity and end pose of the continuum robot is established. Then, the reinforcement learning algorithm is used to take the current pose and target pose of the robot arm as state input, the joint angular velocity of the robot arm as the output action of the agent, and a reasonable reward function is set to guide the robot to move from the initial pose to the target pose. Finally, a simulation system is built in MATLAB, and the simulation results show that the reinforcement learning algorithm successfully performs trajectory planning for the multi-segment continuum robot and controls the end of the continuum robot to move smoothly to the target pose.

Keywords: continuum robot; trajectory planning; reinforcement learning; position and pose control; reward guidance

0 引 言

相比于传统的多关节刚性机器人,连续体机器人凭借灵活性高、适应能力强、交互性好的优点,受到人们越来越多的关注和研究[1]。连续体机器人通过一系列可弯曲的柔性连续臂段,具有更多的自由度,在复杂环境下具有更好的探索能力。经过学术界的广泛研究,在航天[2]、医疗[3]、电力[4]、深海探测[5]、救援[6]等领域,连续体机器人已得到一定的应用,未来有着广阔的应用前景。特别是多段连续体机器人,由若干段可弯曲的柔性臂组成,每一段可向任意方向弯曲成一段圆弧,具有更高的自由度。然而,随着柔性机器人的子段数量的增加,其建模和控制变得更加复杂。

轨迹规划是机器人控制的重要部分。连续型机器人的广泛应用需要进行轨迹规划,以平稳可控的方式使末端执行器从起点运动到终点。胡海燕等[7]针对线驱动连续体机器人的运动学建模问题,分析了单段连续体机器人的完整运动学映射关系,然而针对其提到的两段式连续体机器人并没有给出逆运动学求解方法。韦贵炜等[8]和王浩宇等[9]均采用了同样的方法,由于多段式连续体机器人的逆运动学比较复杂,并没有进行深入分析。为了解决工作空间末端路径点至关节角度的逆映射问题,陈元科等[10]使用粒子群算法进行求解,然而直接求解得到的轨迹是离散的。马

收稿日期:2023-12-18
*基金项目:清华大学横向协作项目(HG2020153)资助

丛俊等[11]在此基础上进一步改进，通过控制关节转动，不断直接使用逆运动学求解并更新当前关节位置，直到末端到达目标位置，规划出连续的关节轨迹。这些方法可以很容易得到连续体机器人的正运动学模型，但是难以解决连续体机器人的逆运动学问题，也没有考虑到末端的姿态控制，特别是逆运动学更加复杂的多段连续体机器人。

速度级运动学可以对多段连续体机器人的末端位姿进行控制，并且其逆运动学可以生成连续轨迹。徐文福等[2]采用比例反馈对连续型空间机械臂的末端位姿进行速度规划，并基于末端速度与关节角速度的运动学模型逆解得到所需的关节角速度。然而该轨迹规划需要对雅可比矩阵求逆，在机器人运动过程中经常会面临奇异问题。Ouyang 等[12]针对连续体机器人的关节角速度逆解过程中的奇异问题，引入了一种奇异点跃迁方法，减小了奇异点对轨迹规划的影响。但是，这样会陷入复杂的模型建立和控制算法设计。

强化学习作为机器学习的一部分，正受到控制学科越来越多学者的研究[13]。强化学习算法是一种无模型方法，不需要复杂的建模与算法设计，非常适用于复杂的连续体机器人。Wu 等[14]采用深度 Q 网络（deep Q-network，DQN）算法对线驱动单段连续体机器人进行控制，使末端可以移动到目标点。Ji 等[15]针对单段线驱动连续体机器人，提出了一种基于多智能体 DQN 的运动控制方法，使机器人末端可以跟踪期望轨迹。但是，DQN 算法难以解决多维动作空间，而深度确定性策略梯度（deep deterministic policy gradient，DDPG）算法可以很好地处理连续动作空间。Li 等[16]针对单段气驱动连续体机器人的运动控制问题，提出了一种基于 DDPG 的控制系统，可以使得机器人末端在二维投影平面可以准确跟踪移动目标。但是单段连续体机器人的自由度有限，通常只能在二维平面内进行轨迹规划。Kargin 等[17]针对三段式连续体机器人在二维平面的轨迹规划问题，使用 DDPG 控制机器人末端运动到目标位置。但是其只考虑了二维平面内的位置控制，没有扩展到三维空间，也没有考虑运动过程中机器人末端的姿态。在机器人的运动控制中，不仅需要考虑末端在目标点的位置，也需要考虑末端的方向，才能更好实现探测或抓取等任务。然而，大多基于强化学习的连续体机器人运动规划研究并未考虑末端的姿态规划。一方面，若连续体机器人的段数少，将导致自由度不足，不能同时控制末端的位置和姿态；另一方面，设计奖励函数时需要将姿态误差表示为一个标量，才能对姿态进行规划。

基于以上问题，本文提出了一种基于 DDPG 强化学习的多段连续体机器人轨迹规划方法。本文的主要创新点在于：1）拓展了连续体机器人的段数，建立三段式连续体机器人的速度级正向运动学模型；2）将 DDPG 强化学习方法应用到三段式连续体机器人的轨迹规划中，可以同时控制末端的位置和姿态，规划出连续的关节空间轨迹。最后，利用三维模型验证了强化学习轨迹规划方法的有效性。

1　连续体机器人模型

1.1　连续体机器人三维模型

本文所指的连续体机器人，灵感来自于章鱼触手、象鼻等生物，通过杆件的弯曲变形形成连续的弧形，来实现末端位姿的移动。连续体机器人由柔性臂杆、支撑盘和驱动绳构成，如图 1 所示，分为 3 段，每段 250 mm。柔性臂杆为弹性材料，负责支撑连续体机器人，并实现关节的弯曲变形。支撑盘负责维持臂杆的近似等曲率弯曲，均匀分布在柔性臂杆上，上面均匀分布 9 个用于穿线的通孔，每 3 个间隔 120°的通孔为一组。驱动绳的一端固定在驱动装置上，另一端穿过基座和支撑盘的通孔，固定在连续体机器人相应段的末端支撑盘上，通过绳子的长度变化控制臂杆的弯曲变形。连续体机器人第 1 段穿过 9 根驱动绳，第 2 段穿过 6 根驱动绳，第 3 段穿过 3 根驱动绳。

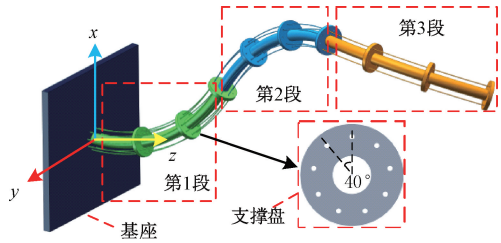


图 1　连续体机器人示意图

1.2　连续体机器人变形描述

考虑连续体机器人的第 i 个臂段（$i=1,2,3$），其弯曲变形如图 2 所示，可以近似等效为一段等曲率的杆件[18]。$o_{i-1}x_{i-1}y_{i-1}z_{i-1}$，$o_ix_iy_iz_i$ 分别为固连于第 i 段首尾两个支撑盘的固连坐标系，原点分别在两端盘心。以下分析仅考虑弯曲变形，忽略扭转变形与拉压变形。

臂段的姿态变换可以由弯曲平面角 φ_i 与弯曲角 θ_i 来描述，即绕 z_{i-1} 轴旋转 φ_i 角，绕 y_{i-1} 轴旋转 θ_i 角，绕 z_{i-1} 轴旋转 $-\varphi_i$ 角，这一弯曲变形后 $o_{i-1}x_{i-1}y_{i-1}z_{i-1}$ 的姿态转为 $o_ix_iy_iz_i$ 的姿态[7]。

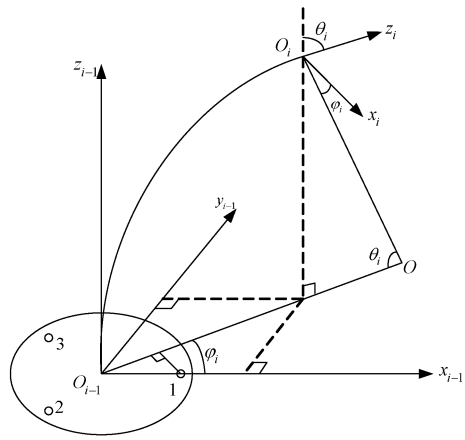


图 2　连续体机器人单段变形描述

定义如下旋转变换:

$$\boldsymbol{R}_z(\varphi_i)=\begin{bmatrix} c_{\varphi_i} & -s_{\varphi_i} & 0 \\ s_{\varphi_i} & c_{\varphi_i} & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{1}$$

$$\boldsymbol{R}_y(\theta_i)=\begin{bmatrix} c_{\theta_i} & 0 & s_{\theta_i} \\ 0 & 1 & 0 \\ -s_{\theta_i} & 0 & c_{\theta_i} \end{bmatrix} \tag{2}$$

式中:c 表示 cos,s 表示 sin。

当弯曲平面角 $\varphi_i=0$ 时,长度为 l 的臂段的弯曲如图3所示,图中 $r_i=l/\theta_i$ 为曲率半径。

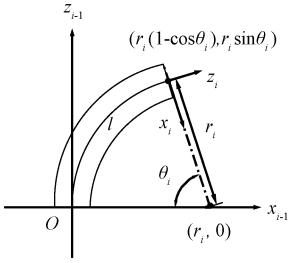


图3 $\varphi_i=0$ 时臂段 i 的变形示意图

该段末端中心在 $o_{i-1}x_{i-1}y_{i-1}z_{i-1}$ 下的坐标可以表示为:

$$\boldsymbol{r}_{i,lcl}=\begin{cases}[(1-c_{\theta_i}) \quad 0 \quad s_{\theta_i}]^{\mathrm{T}} l/\theta_i, & \theta_i\neq 0 \\ [0 \quad 0 \quad l]^{\mathrm{T}}, & \theta_i=0\end{cases} \tag{3}$$

式中:下角标 lcl 表示局部坐标。若将臂段视为一段弧,定义弧长参数 $s\in[0,1]$,$s=0$ 对应弧的首端,$s=1$ 对应弧的末端。在等曲率假设条件下,臂段中弧长参数为 s 处的位置坐标可以表示为

$$\boldsymbol{r}_{i,lcl}(\theta_i,s)=\begin{cases}[(1-c_{s\theta_i}) \quad 0 \quad s_{s\theta_i}]^{\mathrm{T}} l/\theta_i, & \theta_i\neq 0 \\ [0 \quad 0 \quad sl]^{\mathrm{T}}, & \theta_i=0\end{cases}\triangleq \boldsymbol{p}(s\theta_i,sl) \tag{4}$$

当弯曲平面角 $\varphi_i\neq 0$ 时,该段弧长参数为 s 处截面中心在 $o_{i-1}x_{i-1}y_{i-1}z_{i-1}$ 下位置表示为

$$\boldsymbol{r}_{i,lcl}(s)=\boldsymbol{R}_z(\varphi_i)\boldsymbol{p}(s\theta_i,sl) \tag{5}$$

该段弧长参数为 s 处截面在 $o_{i-1}x_{i-1}y_{i-1}z_{i-1}$ 下的姿态可表示为

$$\boldsymbol{R}_{i,lcl}(s)=\boldsymbol{R}_z(\varphi_i)\boldsymbol{R}_y(s\theta_i)\boldsymbol{R}_z(-\varphi_i) \tag{6}$$

绳驱动连续体机器人的运动学分析包括驱动空间(绳长变化)、关节空间(弯曲平面角和弯曲角)、工作空间(末端位姿)的映射关系,如图4所示。上述已推导出关节空间与工作空间的关系,线驱动连续体机器人的驱动空间和关节空间的关系可参考文献[7],不再详述。

1.3 连续体机器人运动学模型

为了给强化学习算法提供训练环境,有必要对连续体机器人进行运动学建模。

假设连续体机器人的基座固定,将与基座固定的第一段首端支撑盘固连坐标系作为惯性坐标系。

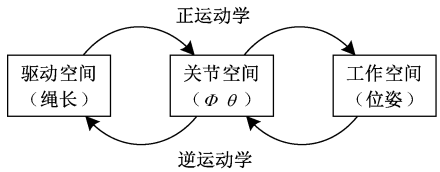


图4 连续体机器人的运动学映射关系

连续体机器人第一段弧长参数为 s 处的绝对位置和绝对姿态可分别表示为:

$$\boldsymbol{r}_1(s)=\boldsymbol{R}_z(\varphi_1)\boldsymbol{p}(s\theta_1,sl) \tag{7}$$

$$\boldsymbol{R}_1(s)=\boldsymbol{R}_z(\varphi_1)\boldsymbol{R}_y(s\theta_1)\boldsymbol{R}_z(-\varphi_1) \tag{8}$$

记 $\dot{\boldsymbol{q}}_1=[\dot{\varphi}_1 \quad \dot{\theta}_1]^{\mathrm{T}}$ 为第一段的虚拟关节向量,对(7)求时间导数可得第1段弧长参数为 s 处的绝对线速度可表示为:

$$\boldsymbol{v}_1(s)=\boldsymbol{R}_{z,\varphi_1}\boldsymbol{p}(\theta_1,s)\dot{\varphi}_1+\boldsymbol{R}_z(\varphi_1)\boldsymbol{p}_\theta(s\theta_1,sl)s\dot{\theta}_1=\boldsymbol{J}_1^v(s)\dot{\boldsymbol{q}}_1 \tag{9}$$

式中:

$$\boldsymbol{J}_1^v(s)=[\boldsymbol{R}_{z,\varphi_1}\boldsymbol{p}(s\theta_1,sl) \quad \boldsymbol{R}_z(\varphi_1)\boldsymbol{p}_\theta(s\theta_1,sl)s]$$

$$\boldsymbol{R}_{z,\varphi_1}=\frac{\partial \boldsymbol{R}_z(\varphi_1)}{\partial\varphi_1}$$

$$\boldsymbol{p}_\theta(s\theta_1,sl)=\frac{\partial \boldsymbol{p}(\theta,sl)}{\partial\theta}\Big|_{\theta=s\theta_1}$$

注:$\boldsymbol{J}_1^v$ 的自变量还包括虚拟关节参数 φ_1,θ_1,但为了表示的简洁,$\boldsymbol{J}_1^v$ 的符号表示忽略了与虚拟关节参数的函数依赖关系。这种表示将用于后续出现的其他符号,并不再赘述。

第1段弧长参数为 s 处的绝对角速度可表示为:

$$\boldsymbol{\omega}_1(s)=\boldsymbol{e}_3\dot{\varphi}_1+\boldsymbol{R}_z(\varphi_1)\boldsymbol{e}_2s\dot{\theta}_1-\boldsymbol{R}_z(\varphi_1)\boldsymbol{R}_y(s\theta_1)\boldsymbol{e}_3\dot{\varphi}_1=\boldsymbol{J}_1^w(s)\dot{\boldsymbol{q}}_1 \tag{10}$$

式中:$\boldsymbol{J}_1^w(s)=[\boldsymbol{e}_3-\boldsymbol{R}_z(\varphi_1)\boldsymbol{R}_y(s\theta_1)\boldsymbol{e}_3 \quad \boldsymbol{R}_z(\varphi_1)\boldsymbol{e}_2s]$,$\boldsymbol{e}_i(i=1,2,3)$ 表示3维单位向量,其第 i 个元素为1,其他元素为0。

由式(9)和(10),第1段弧长参数为 s 处的绝对速度(包括线速度和角速度)可表示为:

$$\boldsymbol{V}_1(s)=\boldsymbol{J}_1(s)\dot{\boldsymbol{q}}_1 \tag{11}$$

式中:$\boldsymbol{J}_1(s)=[(\boldsymbol{J}_1^v(s))^{\mathrm{T}} \quad (\boldsymbol{J}_1^w(s))^{\mathrm{T}}]^{\mathrm{T}}$。

依次建立连续体机器人第2段和第3段的运动学模型。

机器人末端的绝对线速度可表示为:

$$\boldsymbol{v}=\mathbb{J}^v\dot{\boldsymbol{q}} \tag{12}$$

式中:$\mathbb{J}^v$ 是线速度的雅可比矩阵,$\dot{\boldsymbol{q}}=[\dot{\varphi}_1 \quad \dot{\theta}_1 \quad \dot{\varphi}_2 \quad \dot{\theta}_2 \quad \dot{\varphi}_3 \quad \dot{\theta}_3]^{\mathrm{T}}$。

机器人末端的绝对角速度可表示为:

$$\boldsymbol{\omega}=\mathbb{J}^w\dot{\boldsymbol{q}} \tag{13}$$

式中:$\mathbb{J}^w$ 是角速度的雅可比矩阵。

由式(12)、(13),机器人末端的绝对速度可表示为:

$$\boldsymbol{V}=\mathbb{J}\dot{\boldsymbol{q}} \tag{14}$$

式中：$\boldsymbol{V}=[\boldsymbol{v}^{\mathrm{T}}\quad\boldsymbol{\omega}^{\mathrm{T}}]^{\mathrm{T}}$，$\mathbb{J}=[\mathbb{J}^{v\mathrm{T}}\quad\mathbb{J}^{\omega\mathrm{T}}]^{\mathrm{T}}$。

2　强化学习

2.1　强化学习背景

强化学习属于机器学习的范畴，与其他机器学习算法相比，强化学习算法通常是智能体（Agent）与环境（Environment）的交互而不断优化的，其原理示意图如图 5 所示。强化学习算法大多基于马尔可夫决策过程。马尔可夫决策过程可以用元组 (S,A,P,r,γ) 表示，其中 S 是状态(State)的集合，A 是动作(Action)的集合，$P(s'\mid s,a)$ 是状态转移矩阵，即在状态 $s\in S$ 下采取动作 $a\in A$ 之后变为状态 s' 的概率，r 是奖励函数(reward)，由当前状态和采取的动作(或仅有当前状态)决定，γ 是折扣因子，用于计算折扣的奖励值。

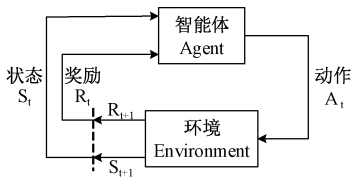


图 5　强化学习示意图

马尔可夫决策过程的执行过程为，在时间步 t，状态为 s_t，根据参数 θ 的策略 $\pi(s_t;\theta)$ 选择并执行动作 a。环境在智能体的动作影响下，状态 s_t 更新为下一时间步的新状态 s_{t+1}，并返回即时奖励 r_t。强化学习算法的目的是找到一个最优策略 π^*，使得回报 $J(\pi_\theta)=E_\pi[\sum_t\gamma^t r(s_t,a_t)]$ 最大化。

在线的强化学习算法在智能体与环境交互过程中，按是否采用经验回放可以分为同策略(on-policy)和异策略(off-policy)。同策略直接使用当前策略的反馈数据进行策略优化，如 TRPO(trust region policy optimization)和 PPO(proximal policy optimization)算法。这样会导致策略侧重于当前策略，限制智能体的策略探索能力，同时学习效率较低，没有充分利用交互过程中有价值的数据信息。因此目前采用得更多的是异策略强化学习算法，将交互数据存入经验回放池，需要时再采样，提高了样本效率。另一方面，DQN 作为异策略算法，只能解决动作空间有限的问题。在大多情况下，机器人系统具有连续动作空间，把无限的动作空间离散为有限个，会牺牲控制精度。DDPG 算法作为热门的强化学习算法，是一种异策略算法，同时可以很好地处理连续动作空间的环境。因此，本文选择使用 DDPG 作为连续体机器人的强化学习轨迹规划算法。

2.2　DDPG 算法

DDPG 是一种基于 Actor-Critic 结构的算法。Actor 是策略网络，负责与环境进行交互，在 Critic 的指导下用策略梯度学习更好的策略。Critic 是价值网络，负责对环境和动作的交互数据进行评估，帮助 Actor 进行策略更新。

DDPG 是 DPG 的一种深度学习扩展版本。DPG 是基于确定性策略梯度的算法，用于处理连续动作空间的问题，并且使用梯度上升方法来更新策略。在高维度的强化学习任务中，价值函数和策略可能非常复杂。万能逼近定理指出，只要隐藏层的层数和神经元个数足够多，神经网络可以用于逼近任何连续函数。因此，DDPG 引入了神经网络，使用深度神经网络近似 Actor 和 Critic 函数。

DDPG 采用随机动作策略进行探索，优化确定性策略，从而提高智能体的学习效率。DDPG 一共有 4 个网络，Actor(策略网络)、Critic(价值网络)以及各自对应的目标网络。与 DQN 不同的是，DDPG 更新目标网络时，不是每隔一段时间将当前的网络直接复制到目标网络，而是采用软更新的方式让目标网络逐渐接近当前网络，从而提高学习的稳定性。为了提高样本利用率，降低交互数据的时间序列相关性对神经网络的影响，DDPG 建立了一个经验回放缓冲区，用来更新策略网络和价值网络。另外，由于 DDPG 采用确定性策略，本身缺乏对环境的探索，需要引入随机噪声帮助 DDPG 进行探索。

DDPG 的算法流程如图 6 所示。首先，初始化智能体和环境，随机初始化 Actor 和 Critic 及目标网络的权重和阈值，初始化经验回放池。然后，设置机器人的初始状态和目标位姿，接收初始观察状态。接下来，在每一个时间步，观察环境状态，根据当前策略选择一个动作，添加探索噪声，并输出动作到环境。智能体执行动作，得到环境反馈的奖励值并观察新状态。然后，智能体从经验回放池中进行随机采样。通过最小化折扣奖励与 Q 值的损失函数更新 Critic，使用采样样本的策略梯度更新 Actor。最后软更新目标网络。智能体不断与环境交互，并训练 Actor 和 Critic 神经网络，直到达到训练终止条件。

3　仿真分析

3.1　参数设置

仿真实验在 MATLAB 仿真平台中进行。机器人控制通常采用位置控制、速度控制或者力矩控制。力矩控制需要进行动力学建模，并且要考虑质量、摩擦、转动惯量等因素，在实际应用中比较困难。如果采用位置控制，则输出的关节角轨迹是不平滑的。因此，选择将关节速度作为强化学习算法的动作空间。强化学习的环境是上述的运动学模型，并在三维环境中建立可视化仿真模型，智能体采用添加噪声的 DDPG 算法。在仿真实验中，将关节速度限定在范围[−0.1,0.1](rad)内。

仿真时间设置为 10 s，保证在规定的时间内连续体机器人末端可以到达目标位姿。环境(即模型)的时间步为 0.001 s，即求解和积分等仿真计算的步长。智能体的时间步为 0.02 s，即观察环境和执行动作的步长。

观察值，即强化学习需要的环境状态，需要包含能够使

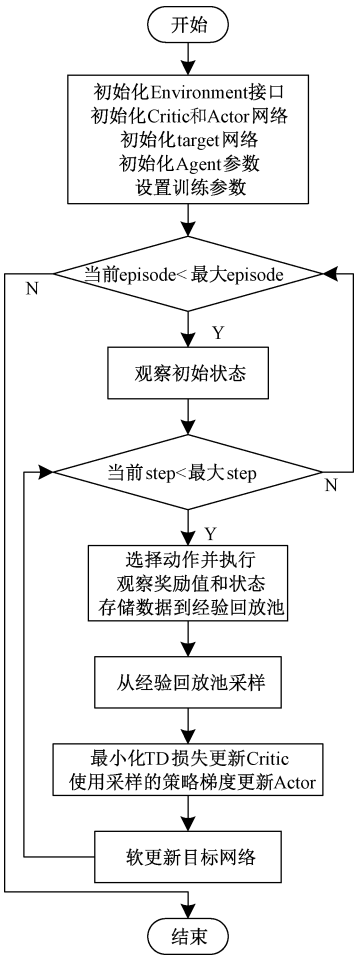


图 6 DDPG 算法流程

智能体做出正确决策的信息。对于固定基座的连续体机器人,状态向量包括:关节角 $\boldsymbol{q}\in R^6$,关节角速度 $\dot{\boldsymbol{q}}\in R^6$,末端执行器的位置 $\boldsymbol{r}_t\in R^3$,线速度 $\boldsymbol{v}_t\in R^3$,目标位置 $\boldsymbol{r}_{goal}\in R^3$,末端执行器的姿态 $\boldsymbol{\psi}_t\in R^3$($\boldsymbol{\psi}_t$ 为四元数 $\boldsymbol{Q}_t$ 对应的欧拉角),角速度 $\boldsymbol{\omega}_t\in R^3$,目标姿态 $\boldsymbol{\psi}_{goal}\in R^3$,末端执行器距离目标的位置绝对误差 $d_r\in R^1$ 和姿态转角误差 $d_\psi\in R^1$。因此,观察值 $\boldsymbol{s}_t\in R^{32}$ 是一个 32 维向量。如果是自由漂浮基座的连续体机器人,还需要考虑基座的位姿及相应的速度。

$$\boldsymbol{s}_t=[\boldsymbol{q}\quad\dot{\boldsymbol{q}}\quad\boldsymbol{r}_t\quad\boldsymbol{v}_t\quad\boldsymbol{r}_{goal}\quad\boldsymbol{\psi}_t\quad\boldsymbol{\omega}_t\quad\boldsymbol{\psi}_{goal}\quad d_r\quad d_\psi] \tag{15}$$

策略函数 Actor 和价值函数 Critic 均采用标准的神经网络组成。Critic 网络如图 7 所示,状态部分包含 2 个隐藏层,动作部分包含 1 个隐藏层,重叠部分包含 1 个隐藏层。Actor 网络如图 8 所示,包含 3 个隐藏层。每个隐藏层有 200 个神经元,激活函数均采用 ReLU 函数。Actor 网络输出值为关节速度,通过 tanh 激活函数缩放至给定的动作范围。

DDPG 算法的超参数设置如下:Actor 和 Critic 网络的学习率为 0.001,目标网络的平滑因子为 0.001,折扣因子为 0.99,采样的批次大小为 128,经验缓冲区大小为 10^6。

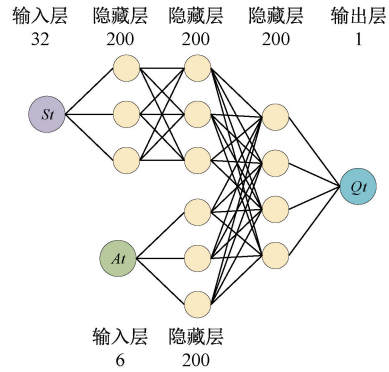


图 7 Critic 神经网络

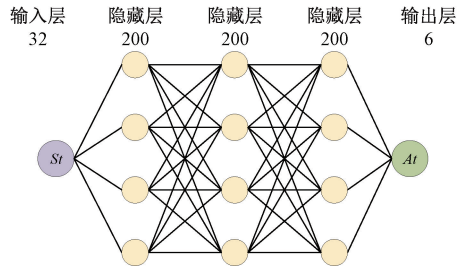


图 8 Actor 神经网络

为了提高 DDPG 对环境的探索,需要对动作添加随机噪声。噪声的概率为 0.4,衰减率为 10^{-5}。

仿真的硬件为 Intel Core i9-9900K CPU 和 NVIDIA TITAN RTX GPU 的电脑,软件为 MATLAB R2023b,训练过程调用了 MATLAB 的强化学习工具箱,并开启 GPU 加速训练。

3.2 奖励函数设计

奖励函数设计是强化学习的难点,只有合适的奖励函数才能引导智能体在训练中完成指定任务。对于多段连续体机器人系统,任务是将机器人末端移动到指定的位置和姿态,进而实现抓取操作。在给定目标位姿时,提前在机器人的工作空间进行采样,保证机器人末端可以运动到给定位姿。

末端的位置误差可以用笛卡尔空间中的直线距离表示。在任意时刻,根据机器人末端的实际位置 $\boldsymbol{r}_t$ 和目标位置 $\boldsymbol{r}_{goal}$,可计算出位置误差:

$$d_r=\|\boldsymbol{r}_t-\boldsymbol{r}_g\| \tag{16}$$

末端的姿态误差可以用姿态四元数和轴角法的转角表示。给定机器人末端的初始姿态四元数 $\boldsymbol{Q}_t$ 和目标姿态四元数 $\boldsymbol{Q}_{goal}$,可根据轴角法计算出姿态误差,即转角误差 d_ψ。

$$\boldsymbol{Q}=\boldsymbol{Q}_{goal}\boldsymbol{Q}_t^{-1} \tag{17}$$

$$d_\psi=2\arccos(a) \tag{18}$$

其中,a 是四元数 $\boldsymbol{Q}$ 的实数部分。

每一回合的终止条件设置为

$$d_r \leqslant 0.005\,\mathrm{m} \text{ and } d_\psi \leqslant 0.01\,\mathrm{rad} \tag{19}$$

奖励函数由 3 部分组成，分别为机器人末端位姿误差惩罚、速度惩罚和到达目标位姿的奖励。奖励函数的数学形式为：

$$r = -c_1 \cdot d_r - c_2 \cdot d_\psi - c_3 \cdot \|\boldsymbol{v}_t\| - c_4 \cdot \|\boldsymbol{w}_t\| - c_5 \cdot \|\dot{\boldsymbol{q}}\| + f \tag{20}$$

其中，$c_i(i=1,\cdots,5)$ 为正系数，f 为机械臂末端到达目标位姿的奖励。

为了使机器人末端执行器尽快到达目标位姿，设置了机械臂末端误差的惩罚项 $-c_1 \cdot d_r - c_2 \cdot d_\psi$。为了使机器人末端可以停在目标位姿附近，设置了末端速度的惩罚项 $-c_3 \cdot \|\boldsymbol{v}_t\| - c_4 \cdot \|\boldsymbol{w}_t\| - c_5 \cdot \|\dot{\boldsymbol{q}}\|$。为了使机器人能够学习能够完成任务的策略，设置了末端达到目标位姿的奖励项 $+f$。在训练中，$c_1 = 100$，$c_2 = 25$，$c_3 = c_4 = c_5 = 1$，$f = 1\,000$。

当机器人末端的位置和姿态均到达目标位姿，则提前结束当前回合并给予奖励项 $+f$。

3.3　仿真结果

连续体机器人的单段长度为 0.25 m，初始关节角假设为 $[0.5,0.5,0.5,0.5,0.5,0.5]^{\mathrm{T}}$（单位为 rad），则相对应的末端初始位置为 $\boldsymbol{r}_0 = [0.407\,8, 0.222\,8, 0.498\,7]^{\mathrm{T}}$（单位为 m），初始姿态为 $\boldsymbol{Q}_0 = [0.731\,7, -0.326\,8, 0.598\,2, 0.000\,0]^{\mathrm{T}}$。期望的目标位置为 $\boldsymbol{r}_{goal} = [0.484\,0, 0.167\,7, 0.059\,8]^{\mathrm{T}}$（单位为 m），目标姿态为 $\boldsymbol{Q}_{goal} = [0.144\,1, -0.328\,4, 0.930\,0, 0.080\,6]^{\mathrm{T}}$。

DDPG 强化学习在训练过程中的奖励如图 9 所示，半透明的曲线是每回合的实际奖励，中间深色曲线是平均奖励。从图中可以看出，在 1 000 回合之前，平均奖励有明显的增长趋势，在 1 000 回合之后奖励值收敛，在 2 500 次以后任务成功率显著增加。训练过程中的 Q 值如图 10 所示。

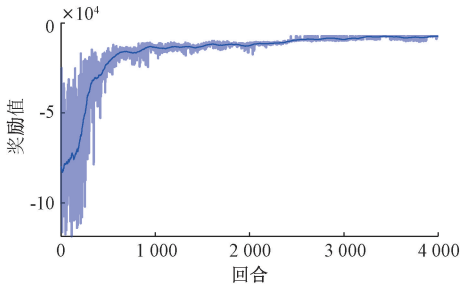


图 9　回合奖励

训练完成后的位姿误差曲线如图 11 和 12 所示，可以看出，DDPG 算法找到了一条良好的末端位姿轨迹，连续体机器人的末端以几乎直线的方式向目标位置移动，同时以比较均匀的速度旋转到目标姿态，最终在 4.7 s 到达目标位姿。

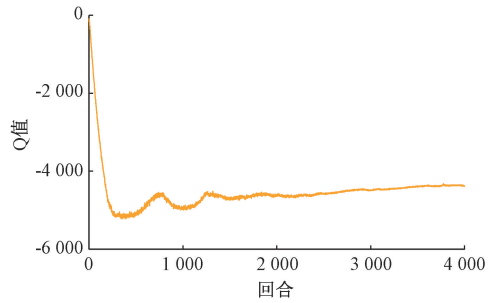


图 10　Q 值

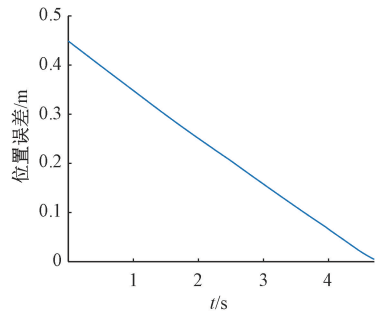


图 11　末端执行器和目标位置的距离

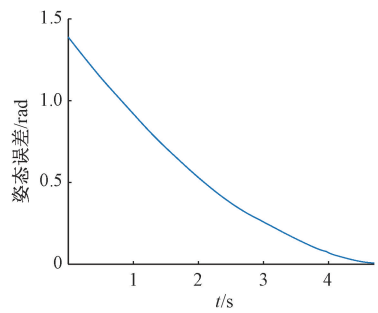
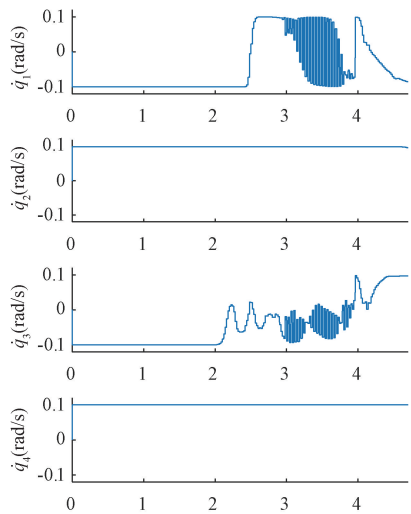


图 12　末端执行器和目标姿态的夹角

强化学习输出的机器人关节速度如图 13 所示，在规定的速度范围 $[-0.1, 0.1]$(rad) 内运动。关节角度如图 14 所示，由关节速度积分得到，生成了一条连续轨迹。



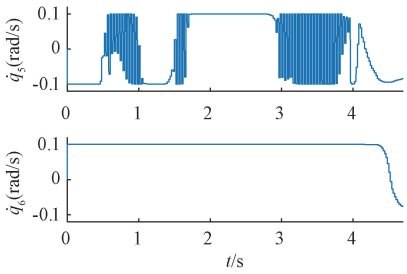


图 13 关节角速度

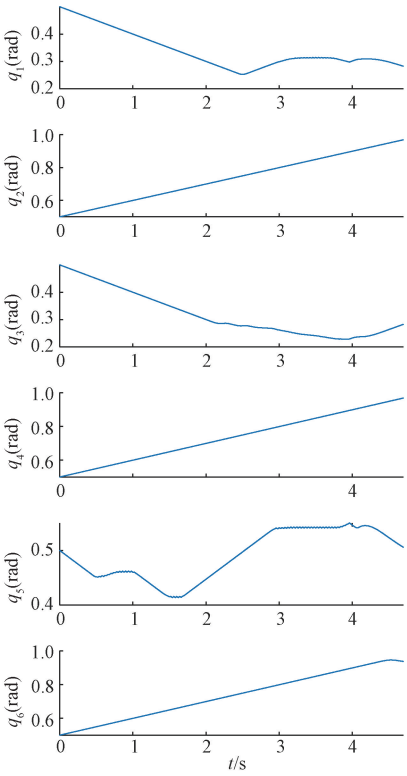


图 14 关节角度

为了验证多段连续体机器人运动学模型的准确性,以及基于 DDPG 强化学习的轨迹规划算法的有效性,在 MATLAB 平台进行了可视化仿真。连续体机器人运动过程如图 15 所示,在基座和每一段末端标记了坐标系,右下角是设定的目标位姿。可以看出,机器人末端坐标系逐渐移动到目标位姿的坐标系,最后完全重合。

3.4 对比分析

为了说明强化学习算法与基于模型的传统算法在轨迹规划的区别,选取基于速度级运动学的轨迹规划方法[12]进行比较。该方法的思路为根据末端速度和雅克比矩阵逆解得到关节角速度,不仅可以同时对末端的位置和姿态进行控制,还可以生成连续的运动轨迹。在本文速度级模型的基础上添加位姿反馈算法[19]和避奇异算法[20],可以根据末端的初始位姿和目标位姿,规划出连续的末端位姿轨迹和关节空间轨迹。

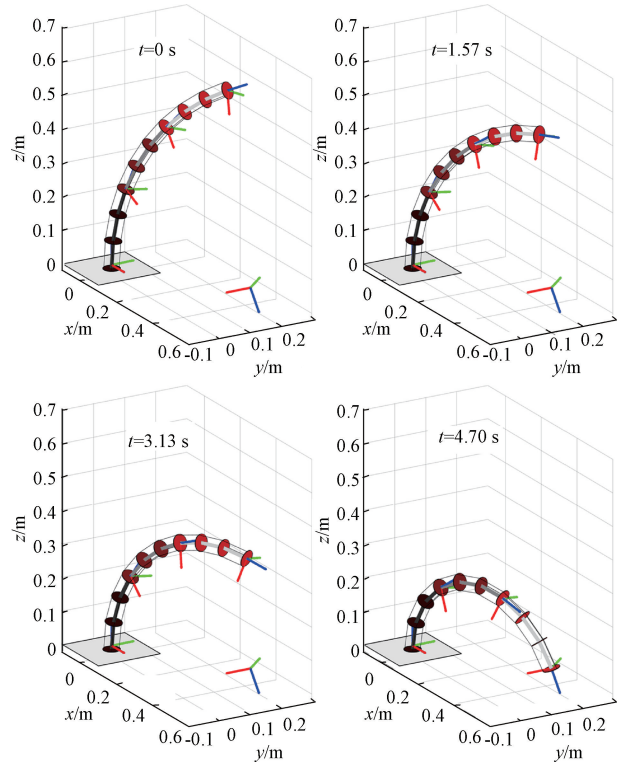


图 15 机械臂运动示意图

假设轨迹规划任务在 10 s 内完成,其中机械臂末端在 0～2 s 匀加速,2～8 s 以最大速度运动,8～10 s 匀减速。机械臂末端距离目标位姿的距离如图 16 和 17 所示,末端执行器以平稳的速度运动到目标位姿。

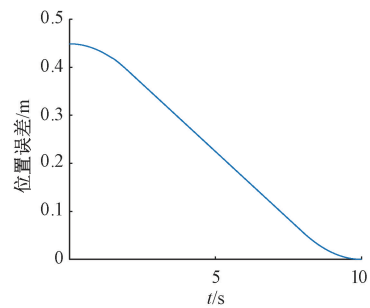


图 16 末端执行器和目标位置的距离

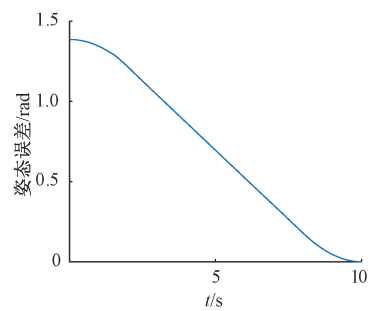


图 17 末端执行器和目标姿态的夹角

基于传统的速度级运动学模型和将规划转化为控制的位姿反馈方法对连续体机器人进行轨迹规划，需要进行复杂的建模和控制器设计。并且由于矩阵求逆的奇异性问题会导致无法求解，需要结合避奇异方法，对关节角速度进行规划。传统方法规划出的关节角速度如图 18 所示，在奇异点的关节角速度达到 4 rad/s 以上，可能会损坏驱动机构，对于连续体机器人的柔性结构会产生较大冲击。相对应的是，采用强化学习进行轨迹规划，通过设置智能体的动作的大小范围，可以在允许的范围内生成关节角速度，关节角速度在 0.1 rad/s 以内。一方面，强化学习轨迹规划简化了建模过程，只需要设计奖励函数，给定初始位姿和末端位姿，即可规划出连续的关节轨迹。一方面，该方法只需要关注终止时刻的目标位姿，避免了运动过程中的奇异问题，显著降低了关节角速度，减小对驱动机构的损坏，保证了多段连续体机器人运动的平稳性。

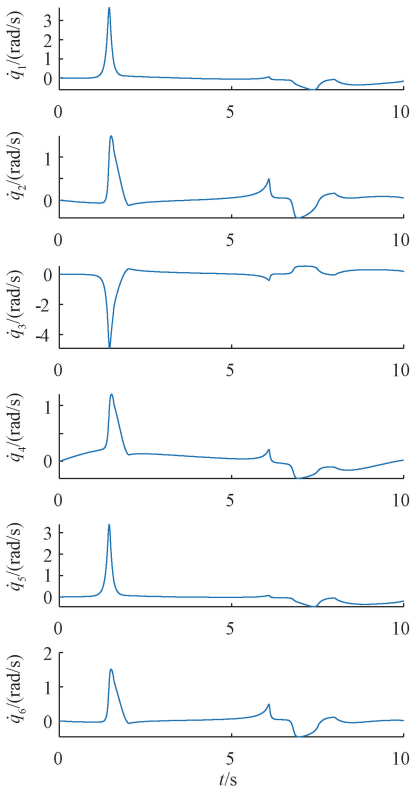


图 18　关节角速度

4　结　　论

在多段连续体机器人的运动学建模的基础上，本文提出了一种基于 DDPG 强化学习的多段连续体机器人轨迹规划方法。通过合理设置相关的状态向量、神经网络结构以及奖励函数，能够成功引导机器人末端向目标位姿移动。经过训练后的算法可以实现对目标位姿的快速规划，不需要复杂的控制算法设计，就能够直接输出关节速度，进而生成连续的关节角度。通过可视化仿真验证了运动学模型和强化学习算法，连续体机器人的末端位姿稳定快速向目标位姿移动。研究成果主要解决了三段式连续体机器人的末端位姿控制问题，并可以生成连续的末端位姿轨迹和关节轨迹。本文展示了基于强化学习的连续体机器人控制的潜力，对强化学习和连续体机器人领域的交叉研究进行了探索。未来将研究提高强化学习应用于连续体机器人的鲁棒性，将训练好的强化学习智能体部署到机器人硬件平台，进行实验验证。

参考文献

[1] HAWKES E W, MAJIDI C, TOLLEY M T. Hard questions for soft robotics[J]. Sci Robot, 2021, 6(53): 6049.

[2] 胡忠华，徐文福，杨太玮，等. 刚柔混合双臂空间机器人抓持-操作协同规划[J]. 宇航学报，2022，43(10)：1311-1321.

[3] RUNCIMAN M, DARZI A, MYLONAS G P. Soft robotics in minimally invasive surgery[J]. Soft robotics, 2019, 6(4): 423-443.

[4] DIAN S, ZHU Y, XIANG G, et al. A novel disturbance-rejection control framework for cable-driven continuum robots with improved state parameterizations[J]. Ieee Access, 2022, 10: 91545-91556.

[5] LI G, CHEN X, ZHOU F, et al. Self-powered soft robot in the Mariana Trench[J]. Nature, 2021, 591(7848): 66-71.

[6] WANG X, ZHANG Q, SHEN D, et al. A novel rescue robot: Hybrid soft and rigid structures for narrow space searching[C]. 2019 IEEE International Conference on Robotics and Biomimetics (ROBIO), IEEE, 2019.

[7] 胡海燕，王鹏飞，孙立宁，等. 线驱动连续型机器人的运动学分析与仿真[J]. 机械工程学报，2010，46(19)：1-8.

[8] 韦贵炜，徐振邦，赵智远，等. 线驱动连续型机械臂设计与运动学仿真[J]. 机械传动，2019，43(11)：32-38,53.

[9] 王浩宇，路铠，王闯，等. 线驱动多关节巡检机械臂运动轨迹[J]. 机械设计与研究，2023，39(2)：57-63.

[10] 陈元科，马飞越，向国菲，等. 用于丝驱动连续体机器人的实用运动学研究[J]. 计算机应用研究，2021，38(10)：3085-3088,3103.

[11] 马丛俊，赵涛，向国菲，等. 基于逆运动学的柔性机械臂末端定位控制[J]. 机械工程学报，2021，57(13)：163-171.

[12] OUYANG X, MENG D, WANG X, et al. Hybrid rigid-continuum dual-arm space robots: Modeling, coupling analysis, and coordinated motion planning[J]. Aerospace

Science and Technology, 2021, 116: 106861.

[13] 张博, 黄山, 张洽芮, 等. 基于强化学习的艾灸机器人温度控制策略研究[J]. 电子测量技术, 2022, 45(24): 60-66.

[14] WU Q, GU Y, LI Y, et al. Position control of cable-driven robotic soft arm based on deep reinforcement learning[J]. Information, 2020, 11(6): 310.

[15] JI G, YAN J, DU J, et al. Towards safe control of continuum manipulator using shielded multiagent reinforcement learning [J]. IEEE Robotics and Automation Letters, 2021, 6(4): 7461-7468.

[16] LI Y, WANG X, KWOK K W. Towards adaptive continuous control of soft robotic manipulator using reinforcement learning [C]. 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS), IEEE, 2022.

[17] KARGIN T C, KOŁOTA J. A reinforcement learning approach for continuum robot control[J]. Journal of Intelligent & Robotic Systems, 2023, 109(4): 1-14.

[18] LIU Z, ZHANG X, CAI Z, et al. Real-time dynamics of cable-driven continuum robots considering the cable constraint and friction effect[J]. IEEE Robotics and Automation Letters, 2021, 6(4): 6235-6242.

[19] LIU Y, YAN W, ZHANG T, et al. Trajectory tracking for a dual-arm free-floating space robot with a class of general nonsingular predefined-time terminal sliding mode [J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2021, 52(5): 3273-3286.

[20] WU J, BIN D, FENG X, et al. GA based adaptive singularity-robust path planning of space robot for on-orbit detection[J]. complexity, 2018:1-11.

作者简介

刘宜成,博士,副教授,主要研究方向为连续体机器人建模与控制、强化学习、空间机器人等。

E-mail:liuyicheng@scu.edu.cn