基于深度强化学习的双足机器人斜坡步态控制方法

吴晓光 [1] 刘绍维 [1] 杨 磊 [1] 邓文强 [1] 贾哲恒 [1]

摘 要 为提高准被动双足机器人斜坡步行稳定性, 本文提出了一种基于深度强化学习的准被动双足机器人步态控制方法. 通过分析准被动双足机器人的混合动力学模型与稳定行走过程, 建立了状态空间、动作空间、episode 过程与奖励函数. 在利用基于 DDPG 改进的 Ape-X DPG 算法持续学习后, 准被动双足机器人能在较大斜坡范围内实现稳定行走. 仿真实验表明, Ape-X DPG 无论是学习能力还是收敛速度均优于基于 PER 的 DDPG. 同时, 相较于能量成型控制, 使用 Ape-X DPG 的准被动双足机器人步态收敛更迅速、步态收敛域更大, 证明 Ape-X DPG 可有效提高准被动双足机器人的步行稳定性.

关键词 准被动双足机器人, 深度强化学习, 步态控制, 步行稳定性

引用格式 吴晓光, 刘绍维, 杨磊, 邓文强, 贾哲恒. 基于深度强化学习的双足机器人斜坡步态控制方法. 自动化学报, 2021, **47**(8): 1976−1987

DOI 10.16383/j.aas.c190547

A Gait Control Method for Biped Robot on Slope Based on Deep Reinforcement Learning

WU Xiao-Guang[1] LIU Shao-Wei[1] YANG Lei[1] DENG Wen-Qiang[1] JIA Zhe-Heng[1]

Abstract In order to improve the walking stability on slope of the quasi-passive biped robot, in this paper, we proposed a gait control method for quasi-passive biped robot based on deep reinforcement learning. By analyzing the hybrid dynamics model and the stable walking process of the quasi-passive biped robot establishing the state space, action space, episode process and reward function. After learning by Ape-X DPG algorithm based on DDPG improvement, quasi-passive biped robot can achieve stable walking in a larger slope range. In the simulation, Ape-X DPG is better than DDPG + PER in both learning ability and convergence speed. Meanwhile, compared with energy shaping controller, the gait convergence of quasi-passive biped robot using Ape-X DPG is more rapid and the basins of attraction is larger, which proves that Ape-X DPG can effectively improve the walking stability of quasi-passive biped robot.

Key words Quasi-passive biped robot, deep reinforcement learning, gait control, walking stability

Citation Wu Xiao-Guang, Liu Shao-Wei, Yang Lei, Deng Wen-Qiang, Jia Zhe-Heng. A gait control method for biped robot on slope based on deep reinforcement learning. *Acta Automatica Sinica*, 2021, **47**(8): 1976−1987

服务机器人融合了机械、控制、计算机、人工智能等众多学科, 在各个领域得到应用, 如足式机器人[1]、水下机器人[2−4]、无人船舶[5]、无人飞行器[6] 等, 是目前全球范围内前沿高科技技术研究最活跃的领域之一. 双足机器人是服务机器人中的一种仿人足式移动机器人, 能够适应街道、楼梯、废墟等复杂的地形环境, 可替代人类从事救援、医疗、勘探、服务等行业. 在双足机器人中, 基于被动步行 (Passive dynamic waking)[7] 理论设计的被动双足机器人, 因结构简单、步态柔顺、能耗低等优点受到广泛研究. 被动双足机器人可充分利用自身动力学特性, 仅依靠重力与自身惯性便能沿斜坡向下行走. 然而, 被动双足机器人在行走过程中因缺乏主动控制, 存在步行稳定性差、抗扰动能力弱等不足. 为弥补这些不足, 研究人员通过对被动双足机器人部分关节施加控制, 研发出准被动双足机器人[8], 提升了双足机器人的步态控制能力.

为进一步提高准被动双足机器人步行稳定性, 步态控制方法的研究逐步成为准被动双足机器人研究领域的重点方向, 现有的控制方法包括神经网络[9]、延时反馈控制[10−11]、能量成型控制[12−13]、强化学习[14] 等. 其中, 强化学习 (Reinforcement learning, RL) 因易于实现、适应性好、无需先验知识等优点而得到广泛应用. Tedrake 等[15] 利用随机策略梯度 (Stochastic policy gradient, SPG) 算法实现无膝双足机器人 Toddler 的步态控制, 使其能够在不平整

收稿日期 2019-07-23 录用日期 2020-01-09
Manuscript received July 23, 2019; accepted January 9, 2020
国家自然科学基金 (61503325), 中国博士后科学基金 (2015M581316) 资助
Supported by National Natural Science Foundation of China (61503325), China Postdoctoral Science Foundation under Grants (2015M581316)
本文责任编委 程龙
Recommended by Associate Editor CHENG Long
1. 燕山大学电气工程学院 秦皇岛 066004
1. School of Electrical Engineering, Yanshan University, Qinhuangdao 066004

路面上行走. Hitomi 等[16] 则将 SPG 应用于一种圆足有膝双足机器人的控制中, 实现机器人在 [0.02, 0.04] rad 斜坡范围上的稳定行走, 并提升了机器人对外界扰动的鲁棒性. Ueno 等[17] 采用改进的行动者−评论家 (Actor-critic, AC) 算法提高了具有上肢双足机器人的步行稳定性, 使机器人在 20 组实验中完成 19 次稳定行走. 然而, 上述算法均受 RL 的结构、学习能力的制约, 存在样本利用率低、学习不稳定、算法不易收敛等缺陷, 严重限制了 RL 对机器人步态的控制能力.

近年来, 结合强化学习和深度学习的深度强化学习 (Deep reinforcement learning, DRL) 快速发展, 迅速成为人工智能领域的研究热点[18]. DRL 利用深度学习的优点克服传统 RL 中的缺陷, 广泛应用于自动驾驶[19−20]、自然语言处理[21−23] 等领域, 并被引入到双足机器人的步态控制研究中. 在主动双足机器人中, 赵玉婷等[24] 利用深度 Q 网络 (Deep Q network, DQN)[25] 算法, 有效抑制了机器人在非平整地面行走时姿态角度的波动. 在准被动双足机器人中, Kumar 等[26] 将有膝双足机器人视为智能体, 利用深度确定性策略梯度 (Deep deterministic policy gradient, DDPG)[27] 算法, 实现机器人长距离的行走. 此外, DRL 研究中也常将双足机器人作为控制对象, 如 MuJoCo[28−29] 中的 2Dwarker 模型、Roboschool[30] 中的 Atlas 模型等.

由于准被动双足机器人步态稳定的判别较为困难, DRL 在控制准被动双足机器人时, 通常以行走的更远为目的, 忽略了机器人步行稳定性、柔顺性等因素, 这导致 DRL 控制下机器人步态与稳定步态之间存在较大的差异. 针对此问题, 结合传统 RL 在准被动双足机器人步态控制方面的不足, 本文提出了一种基于 DRL 的准被动双足机器人步态控制方法, 实现较大斜坡范围 ([0.04, 0.15] rad) 下的机器人不稳定步态控制, 使机器人能够抑制跌倒并快速恢复至稳定步态, 达到提高机器人步行稳定性的目的: 1) 建立准被动双足机器人动力学模型, 确立机器人的状态空间与动作空间. 2) 针对 DDPG 的不足, 基于优先经验回放 (Prioritized experience replay, PER)[31] 机制, 引入分布式优先经验回放 (Distributed prioritized experience replay, DPER)[32] 结构, 建立高效的机器人步态控制方法 — Ape-X DPG 算法. 3) 基于准被动双足机器人的行走特性设计的 Episode 过程, 结合机器人步态变化与缩放动作构建的奖励函数, 为 Ape-X DPG 的高效学习提供支撑. 4) 通过仿真实验, 对 Ape-X DPG 的学习能力和步态控制能力进行测试分析, 验证步态控制方法的有效性.

1　双足机器人动力学模型

1.1　动力学模型建立

本文以直腿前向圆弧足机器人作为研究对象, 构建其动力学模型, 机器人物理模型如图 1 所示. 机器人由连接在髋关节 H 处的两条完全一致的刚性直腿组成, 被动步行时具有两个自由度, 分别位于支撑点 s 与髋关节 H 处, 记为 θ_1 与 θ_2. 为实施主动控制, 在机器人髋关节与两腿的踝关节处设有电机. 对机器人行走过程做运动简化[33], 可将行走过程划分为摆动阶段和碰撞阶段, 机器人被动步行过程如图 2 所示.

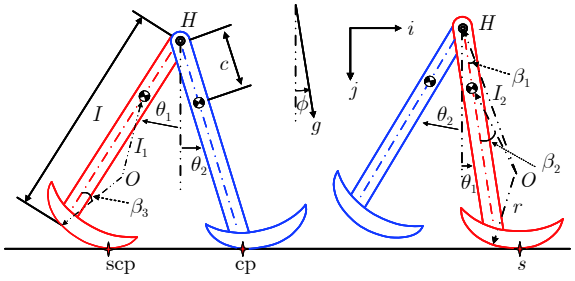


图 1　机器人模型示意图

Fig. 1　Sketch of the biped model

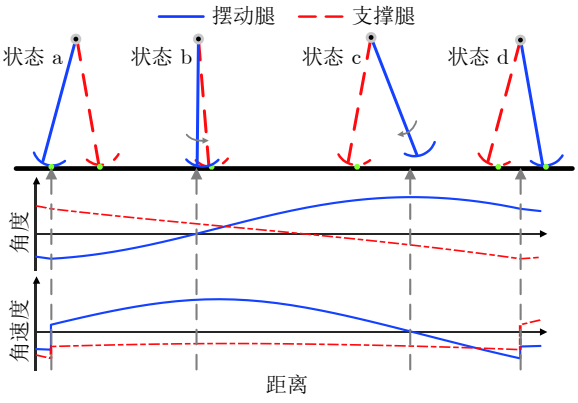


图 2　被动步行过程

Fig. 2　Passive dynamic waking process

图 2 中, 状态 a 至状态 d 前为摆动阶段. 此阶段, 机器人支撑腿绕支撑点 s 做倒立摆运动, 摆动腿离地并绕髋关节 H 做单摆运动, 运动中忽略摆动腿的擦地现象, 由 Lagrange 法推导摆动阶段动力学方程:

$$M(q)\ddot{q} + H(q,\dot{q}) = u(t) \tag{1}$$

其中, q 为姿态向量 $[\theta_1, \theta_2]$; $M(q)$ 为 2×2 正定质

量惯性矩阵; $H(q,\dot{q})$ 为重力、离心力和哥氏力之和; $\mu(t)=[\mu_{st},\mu_{sw}]$ 为控制力矩集合, μ_{st}、μ_{sw} 分别为支撑腿踝关节与摆动腿髋关节处的电机力矩, 当 $\mu(t)=[0,0]$ 时机器人处于被动步行状态.

状态 d 时刻, 机器人处于碰撞阶段. 此阶段, 机器人摆动腿在碰撞点 cp 处与地面发生瞬时完全非弹性碰撞, 碰撞前后 $\dot{\theta_1}$、$\dot{\theta_2}$ 发生突变, 碰撞后, 支撑腿与摆动腿间角色交换, 满足:

$$\begin{cases}\theta_1^+=\theta_2^--2\beta_3\\ \theta_2^+=\theta_1^-+2\beta_3\end{cases}\tag{2}$$

其中, β_3 为前向补偿角, "–"、"+"分别表示碰撞前瞬间和碰撞后瞬间. 由于碰撞前后机器人关于碰撞点 cp 处角动量守恒, 碰撞后摆动腿关于髋关节 H 处角动量守恒, 可得到碰撞阶段动力学方程:

$$Q^-(q)\dot{q}^-=Q^+(q)\dot{q}^+\tag{3}$$

其中, Q^- 与 Q^+ 可由碰撞前后角动量守恒推导得到. 联立式 (1)~(3) 完成机器人行走过程的混合动力学模型建立.

1.2 状态空间与动作空间

当机器人作为智能体时, 其受控行走过程可用马尔科夫决策过程 (Markov decision processes, MDP) 描述. 通常, MDP 可记为四元数组 (S,A,P,R). 其中, S 为智能体状态空间, A 为智能体动作空间, P 为状态转移函数, R 为奖励函数. 本文中, 将机器人的状态空间定义为 $S=[x,\phi]$, 其中, $x=[\theta_1,\dot{\theta_1},\dot{\theta_2}]$ 为机器人起始状态, ϕ 为斜坡坡度; 令机器人动作空间为 $A=\mu_{sw}$, 在机器人摆动摆动阶段中 μ_{sw} 恒定, 可有效防止摆动腿在行走中抖动, 保证步态的柔顺; 由于 μ_{st} 空间范围更为广泛但对本文所选取的坡度范围下无明显的控制提升, 因此令 $\mu_{st}\equiv 0$ 即锁死踝关节, 以减少训练耗时与控制能耗. 因此在第 t 步时, 机器人的行走过程可以描述为: 状态 s_t 的机器人执行 DRL 选择的动作 a_t, 根据 P 迁移至状态 S_{t+1}, 并通过 R 得到奖励值 $r_t(s_t,a_t)$.

为减少分析参数, 选取足地碰撞后瞬时时刻的机器人状态空间为庞加莱截面, 则机器人状态的转换可利用庞加莱映射 f 实现, 满足:

$$x_{t+1}=f(x_t)\tag{4}$$

若存在状态 x, 满足 $x=f(x)$, 称状态 x 为不动点, 此时机器人步态即为稳定步态. 结合 MDP 可知, 以步态稳定为目标时, DRL 需选择动作使机器人快速到达不动点, 以获得更高的奖励值.

2 深度确定性策略梯度算法

DDPG 是基于确定性策略梯度 (Deterministic policy gradient, DPG)[34] 改进的一种离线、无模型 DRL 算法, 适用于连续动作空间问题. 采用 DDPG 控制机器人行走, 可以使机器人获得更准确的控制, 加快步态的收敛速度. 进一步利用 PER 替代 DDPG 原有的样本抽取机制, 可提高样本利用率, 改善 DDPG 的学习能力.

2.1 算法结构

在 DDPG 中, 分别使用策略神经网络 μ 与价值神经网络 Q 表示 DPG 与状态动作值函数, 并组成 AC 算法. 其中, μ 为 Actor, 当机器人状态为 s_t 时, μ 选择动作 a_t 的过程为:

$$a_t=\mu(s_t|\theta^\mu)+N_t\tag{5}$$

其中, θ^μ 为 μ 的神经网络参数; N_t 为动作扰动, 由扰动函数 N 提供, 用以在学习过程中探索环境. 机器人在执行动作 a_t 后, 结合返回的 s_{t+1} 与 r_t, 将其结合 s_t 与 a_t 组成样本 $[s_t,a_t,r_t,s_{t+1}]$ 存入样本池. 价值网络 Q 作为 Critic, 用以逼近状态动作值函数:

$$q=Q(s_t,a_t|\theta^Q)\tag{6}$$

其中, θ^Q 为 Q 的神经网络参数.

为稳定学习过程, DDPG 借鉴 DQN 中的目标网络结构, 构建目标策略网络 μ' 与目标价值网络 Q', 并在目标网络中引入缓慢更新策略:

$$\begin{cases}\theta^{Q'}=\tau\theta^Q+(1-\tau)\theta^{Q'}\\ \theta^{\mu'}=\tau\theta^\mu+(1-\tau)\theta^{\mu'}\end{cases}\tag{7}$$

其中, $\theta^{Q'}$、$\theta^{\mu'}$ 分别为 Q'、μ' 的神经网络参数; τ 控制着 $\theta^{Q'}$、$\theta^{\mu'}$ 的更新幅度, 通常取 $\tau<<1$. 对于策略网络 μ 与价值网络 Q, 则使用经验回放 (Experience replay, ER) 机制从样本池中随机抽取训练样本集进行离线训练. 结合目标网络 μ' 和 Q', 对于训练样本集 I, Q 的损失函数和 μ 的梯度更新分别为:

$$L(\theta^Q)=\frac{1}{I}\sum_i^I(r_i+\gamma Q'(s_{i+1},\mu'(s_{i+1}|\theta^{\mu'})|\theta^{Q'})-Q(s_i,a_i|\theta^Q))^2\tag{8}$$

$$\nabla_{\theta^\mu}J\approx\frac{1}{I}\sum_i^I\nabla_{a_i}Q(s_i,a_i|\theta^Q)\nabla_{\theta^\mu}\mu(s_i|\theta^\mu)\tag{9}$$

式中, γ 为奖励折扣; Q' 与 μ' 通过降低 Q 的变化幅度, 抑制训练中 Q 和 μ 的网络震荡, 达到稳定算法学习过程的目的, DDPG 中的神经网络训练过程如图 3 所示.

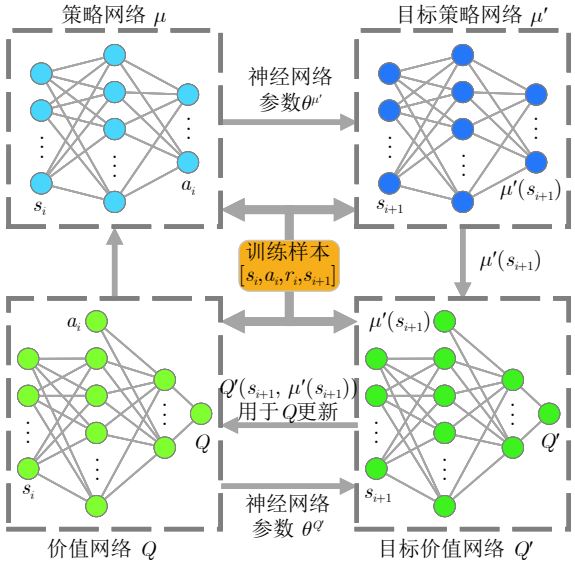


图 3　DDPG 中神经网络训练过程

Fig. 3　The neural network training process in DDPG

2.2　优先经验回放策略的引入

虽然 ER 能够打破样本间的相关性, 满足 DDPG 中神经网络的离线训练要求, 但 ER 并不能判断其抽取样本的训练价值, 导致 DDPG 无法充分利用样本. 为改善这一不足, 采用 PER 替代 ER, 以提升高价值样本利用率.

PER 使用时间差分 (Temporal difference, TD) 误差[35] 表示样本的价值. 令样本 TD 误差绝对值越大时价值越高, 则对于样本 i, 在 DDPG 中的 TD 误差为:

$$D_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1})) - Q(s_i, a_i) \tag{10}$$

将样本池中的样本按 TD 误差绝对值进行降序排列, 建立样本的抽取优先级:

$$p_i = \frac{1}{rank(i)} \tag{11}$$

其中, $rank(i)$ 为样本 i 排序后的队列序号, 最高优先级 $p_{max} = 1$, 即价值越高优先级越高. 相比于直接使用 TD 误差作为抽取依据, p_i 能够更好地抑制噪声样本的影响, 此时样本 i 被抽取概率为:

$$P_i = \frac{p_i^\alpha}{\sum_k p_k^\alpha} \tag{12}$$

其中, k 为样本总量; $\alpha \in [0,1]$ 可调节高价值样本在样本集中的比例, 确保样本集内的样本多样性, $\alpha = 0$ 时即为随机抽取. 同时, PER 中还使用重要性采样权重 (Importance-sampling weights, IS) 对频繁回放高价值样本造成的影响进行纠正, 确保学习过程的稳定, 样本 i 的 IS 值可表示为:

$$w_i = \left(\frac{1}{kP(i)}\right)^\beta \tag{13}$$

其中, $\beta \in [0,1]$ 可控制纠正的程度, 在价值网络 Q 的损失函数中加入 IS 值, 损失函数更新为:

$$L(\theta^Q) = \frac{1}{I}\sum_i^I w_i(r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1}|\theta^{\mu'})|\theta^{Q'}) - Q(s_i, a_i|\theta^Q))^2 \tag{14}$$

3　基于 Ape-X DPG 的步态控制方法

基于 PER 的 DDPG 通过改变样本抽取机制进而改善算法学习能力, 但其学习过程中训练与交互需顺序交替执行, 限制了样本的采集速度, 增加了学习时间. 为此, 本文在基于 PER 的 DDPG 的基础上引入 DPER 结构, 形成 Ape-X DPG 算法, 整体结构如图 4 所示.

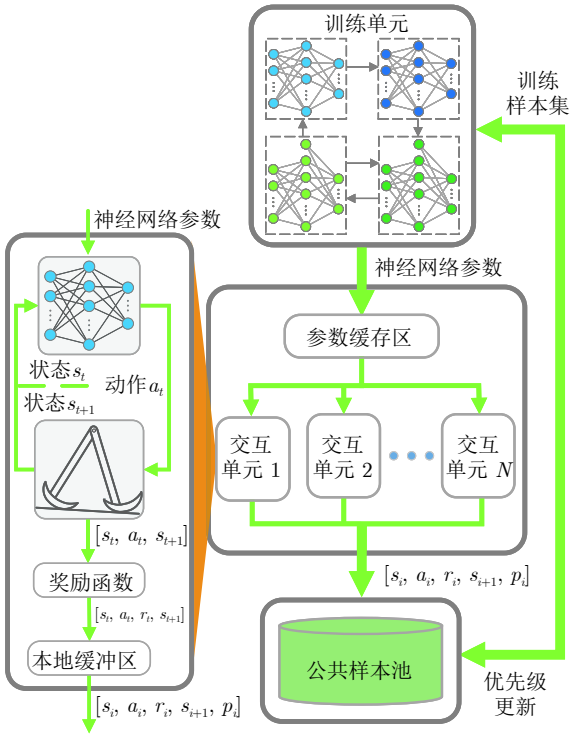


图 4　Ape-X DPG 算法结构

Fig. 4　The structure of Ape-X DPG

3.1　Ape-X DPG 结构

如图 4 所示, Ape-X DPG 主要由三部分组成:

1) 交互单元. 交互单元负责收集机器人的行走样本, 可依据计算机性能部署多个, 各单元间相互独立. 交互单元由本地 DDPG、本地样本池、机器人

交互环境组成. 其中, 本地 DDPG 控制机器人的行走, 其从参数缓冲区中获得网络参数; 本地样本池用于样本的缓存, 当样本量超过存储上限时, 计算样本初始优先级并送入公共样本池.

2) 公共样本池. 公共样本池负责存储交互中产生的所有样本, 同时使用 PER 为训练单元抽取训练样本集. 为减少样本在排序与抽样时的计算消耗, 公共样本池采用二叉树结构.

3) 训练单元. 训练单元利用样本集不断训练学习. 训练单元本身不直接参与机器人的交互, 但每次训练后, 其会将训练后参数存入参数缓冲区中, 并更新训练样本在公共样本池中的优先级.

为简化结构, 本文将交互单元中的本地 DDPG 使用策略神经网络 μ 进行替代, 称为本地 Actor. 简化后, 样本的初始优先级均为 $p_{max}=1$, 此时使用 PER 的公共样本池会优先抽取传入的新样本, 使训练单元更重视最新样本的处理.

Ape-X DPG 通过上述三部分的并行执行, 将 DDPG 的交互与训练相分离, 从而有效缩短学习时间. 同时, 多个交互单元的部署, 极大地提升样本的收集速度, 而交互单元间的相互独立, 使得不同交互单元可以采用不同扰动函数 N, 增强了算法探索环境的能力, 简化后的 Ape-X DPG 过程如下所述:

算法 1. 交互单元 n

1) 由参数缓冲区获得本地 Actor 的神经网络参数 θ_i^{μ}

2) 初始化本地样本池 K_n、随机扰动函数 N_n, 设置 K_n 上限 $Size$

3) for $e=1$ to M:

4) 本地 Actor 控制机器人完成一次 Episode

5) if $K_n > Size$:

6) 对 K_n 中的样本赋予优先级 $p_{max}=1$

7) 将 K_n 中的样本存入公共经验样本池 K, 并清空 K_n

8) 从参数缓冲区中更新神经网络参数 θ_n^{μ}

9) end if

10) end for

算法 2. 训练单元

1) 随机初始化价值网络 Q 和策略网络 μ 的网络参数 θ^{Q}、θ^{μ}, 并传入参数缓冲区

2) 初始化目标网络网络参数 $\theta^{Q'} \leftarrow \theta^{Q}$、$\theta^{\mu'} \leftarrow \theta^{\mu}$

3) for $t=1$ to T:

4) 公共样本池 K 使用 PER 抽取训练样本集 I, 并取出对应的 IS 值 w_i

5) 通过最小化损失函数式 (14) 对 Q 进行更新

6) 依据梯度更新式 (9) 对 μ 进行更新

7) 目标网络获得参数更新

8) 将 θ^{μ} 传入参数缓冲区

9) 计算各训练样本最新的 TD 误差 D_i

10) 根据 D_i 更新训练样本在 K 中的优先级

11) end for

3.2 双足机器人交互过程

为使 Ape-X DPG 习得高效的步态控制策略, 限定一次 Episode 中最大行走步数为 10 步. 同时, 为模拟机器人多样的不稳定步态, 随机选择机器人的初始状态 s_1, 交互单元 n 中 1 次 Episode 的过程如图 5 所示.

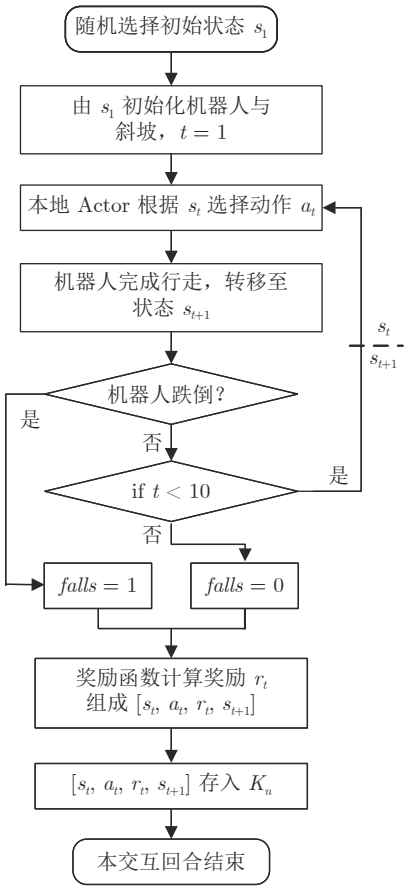


图 5 交互单元 n 中 Episode 过程

Fig. 5 Episode process in interaction unit n

图 5 中, $falls$ 用于标识机器人在本次 Episode 的完成状态, 若在 Episode 中机器人跌倒, 则将 Episode 中的各步标记为 $falls=1$;若机器人完成 10 步行走, 则各步标记为 $falls=0$.

3.3 奖励函数设计

不动点能够表征机器人稳定行走时的状态, 常被用于奖励函数的设计. 但由于不动点求解困难且随 ϕ 的变化而变化, 因此其不适合较大斜坡范围下

的奖励函数设计. 当机器人状态处于不动点时, 机器人步态单一且无需外力矩作用, 因此奖励函数可设计为:

$$r(s_t,a_t)=\begin{cases}\exp(-\Delta^2-{a_r}^2), & falls=0\\ -1, & falls=1\end{cases}\tag{15}$$

其中, $\Delta=4\|x_{t+1}-x_t\|_2$ 表示机器人在庞加莱截面上的步态变化, x_t、x_{t+1} 分别从样本中的 s_t、s_{t+1} 获得; $a_r=25\,|a_t|$ 为缩放后的动作 a_t. 当 $falls=1$ 时, 机器人获得奖励值−1. 当 $falls=0$ 时, 奖励函数利用 Δ 与 α_r 替代不动点评价机器人步态稳定程度, 奖励函数空间如图 6 所示:

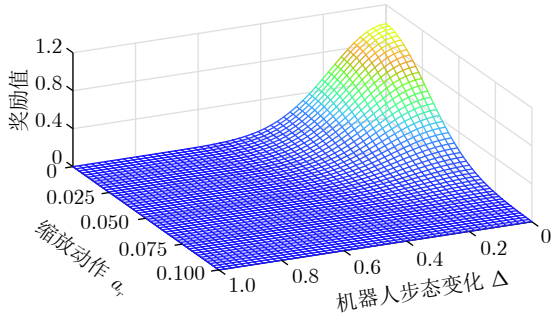


图 6　$falls=0$ 时的奖励函数空间

Fig. 6　Landscape of the reward function when $falls=0$

图 6 中, 奖励函数空间整体呈现单调变化趋势, 当 Δ 与 a_r 均趋近 0 时将步态视为稳定, 给予机器人高奖励值. 在奖励函数中引入动作 a_t, 可以引导 Ape-X DPG 选择较小的动作调节机器人步态, 提高机器人能效, 同时减小对稳定步态的扰动.

4　仿真实验

4.1　仿真实现细节

本文通过 Python 与 Matlab 的联合仿真对 Ape-X DPG 的学习与控制能力进行验证. 其中, Python 负责 Ape-X DPG 的实现; Matlab 负责机器人的动力学仿真; Python 与 Matlab 间通过 Matlab Engine 进行通信. 为保证结果的一致性, 图像均使用 Matlab 进行绘制.

在 Matlab 中, 机器人物理参数设置如表 1 所示. 为更好地检验算法控制能力, 仿真实验中机器人采用稳定性较差的对称圆弧足[36], 此时 $\beta_3=0$、碰撞后 $\theta_1=-\theta_2$. 基于机器人的步态运动特征, 限定初始时的状态空间 S_1 范围: $\theta_1\in[0.02,0.6]$、$\dot{\theta}_1\in[-2,-0.08]$、$\dot{\theta}_2\in[0,6]$、$\phi\in[0.04,0.15]$, 并限定动作空间 A 中的 $u_{sw}\in[-0.3,0.3]$.

表 1　机器人符号及无量纲参数

Table 1　Symbols and dimensionless default values of biped parameters

参数	符号	数值
腿长	I	1
腿部质心	m_1	1
髋关节质心	m_2	2
足半径	r	0.3
腿部质心与圆弧足中心距离	I_1	0.55
髋关节与圆弧足中心距离	I_2	0.7
髋关节到腿部质心距离	c	0.15
腿部转动惯量	J_1	0.01
重力加速度	g	9.8

在 Python 中, 采用 Tensorflow[37] 实现的 Ape-X DPG, 交互单元、公共样本池分配于不同 CPU 核心中, 训练单元则分配至 GPU 中. 训练单元中, Q 和 μ 使用全链接神经网络, 均有 4 个隐藏层, 各层单元数分别为 100、300、200、50, 使用 ReLU 激活函数. 输入层单元数由状态空间 S 决定, 且 Q 在第 3 隐藏层中接收对应动作 a. 对于输出层激活函数, μ 使用 tanh 激活函数, 而 Q 不使用激活函数. 在训练过程中, μ 和 Q 的学习率分别设置为 10^{-4}、10^{-3}, 使用 Adam 算法[38] 更新. μ' 和 Q' 更新参数 τ 为 10^{-4}.

公共样本池存储上限为 5×10^5, PER 中参数设置分别为 $\alpha=0.7$、$\beta=0.4$. 交互单元中, 本地 Actor 结构与 μ 相同, 本地样本池存储上限为 10^3.

4.2　学习过程

由于交互单元数量与扰动函数 N 设置均影响着算法性能, 本文通过 3 组不同交互单元数量的 Ape-X DPG 和 1 组基于 PER 的 DDPG 进行仿真实验, 对比算法的学习能力与收敛速度. 在学习开始后, 当公共样本池样本总数到达 2.5×10^5 时启动训练单元, 当各采集单元均完成 20 000 次 Episode 时结束学习. 各组仿真实验中 N 分配与训练单元启动后的学习耗时如表 2 所示, 表中 0, 1, 2 表示该组算法中使用对应 N 的交互单元数. 训练单元启动后各算法奖励曲线如图 7 所示.

图 7 中, 曲线表示各组算法中交互单元对应 Episode 奖励的平均值. 其中, 点线为 DDPG 的奖励值, 可以看出其学习速度缓慢且学习过程存在明显的震荡, 虽然与 2 交互单元 Ape-X DPG 最终的平均奖励值相差不大, 但 Ape-X DPG 的收敛更早且过程也更加稳定. 同时, 由于 Ape-X DPG 中交互与训练并行运行, 因此在相同条件下, Ape-X DPG

表 2 扰动函数 N 分配与学习耗时

Table 2 Noise function N settings and learning time

算法	高斯扰动	O-U 扰动	网络参数扰动[39]	耗时
DDPG	0	1	0	6.4 h
2 交互单元	1	1	0	4.2 h
4 交互单元	2	1	1	4.2 h
6 交互单元	2	2	2	4.3 h

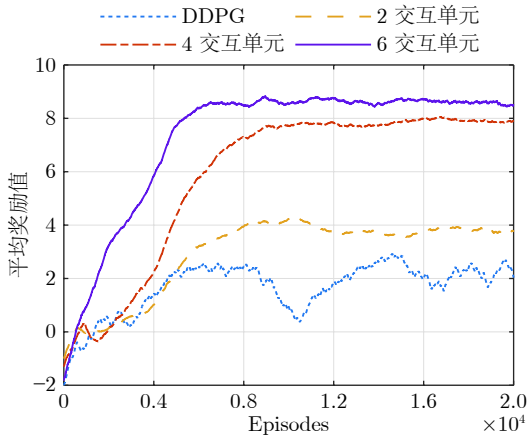


图 7 平均奖励值曲线

Fig. 7 The curves of the average reward

的学习耗时显著低于 DDPG. 对于三组 Ape-X DPG, 其在收敛后的平均奖励值大小与交互单元数量成正比, 且由于各交互单元独立运行, 使得三组算法在执行固定交互次数时学习整体耗时差异较小.

4.3 步态控制分析

为测试 Ape-X DPG 的步态控制能力, 本文选择能量成型控制作为对比控制算法. 当机器人在斜坡上被动行走时, 摆动阶段中重力所做功转化为系统动能, 若碰撞阶段中这部分能量被精确消耗, 则机器人能量变化形成平衡, 可实现稳定行走; 若能量无法被精确消耗, 则会导致机器人行走的不稳定. 能量成型控制利用上述过程, 将机器人不动点处能量总值 E_{target} 作为参考, 通过 $\mu_t = [\mu_{st},\ \mu_{sw}]$ 的作用, 使机器人不稳定初始能量 E_t 快速收敛至 E_{target}, 进而实现步态的调整, 有控制率:

$$\mu_t = -\lambda_t(E_t - E_{target})\dot{q} \tag{16}$$

其中, λ_t 为自适应系数, 具体为:

$$\lambda_t = \frac{\lambda_{t-1}}{1 - \tanh(\xi \upsilon_{t-1})} \tag{17}$$

其中, ξ 为可调阻尼系数, υ_{t-1} 为机器人第 t–1 步时步态周期变化, 表示为:

$$\upsilon_{t-1} = \ln\left|\frac{T_{t-3} - T_{t-2}}{T_{t-2} - T_{t-1}}\right| \tag{18}$$

其中, T_{t-1} 为机器人第 t–1 步的行走时间. λ_t 能够依据步态不稳定程度对力矩大小进行调整, 加快步态收敛速度.

不失一般性, 从初始状态空间中随机抽取 2000 组作为测试集. 能量成型控制选取参数 $\xi = 0.8$, 初始 λ_t 为 $\lambda_1 = 0.01$, 对于初始阶段所需的 T_{t-1} 等, 使用稳定步态行走时间补充. DDPG 与 Ape-X DPG 使用上节训练所得参数, 在机器人行走 15 步时检测步态是否稳定, 各算法稳定行走次数如图 8 所示.

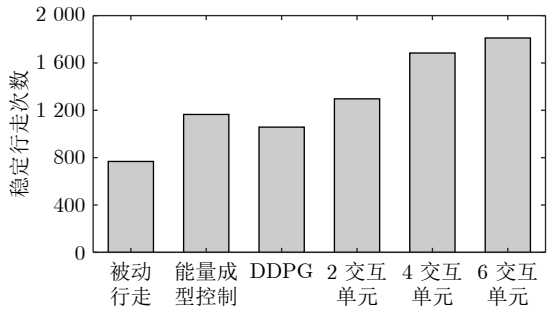


图 8 测试集稳定行走次数

Fig. 8 Stable walking times in test

图 8 中, 能量成型控制同 DDPG、2 交互单元 Ape-X DPG 的稳定行走次数接近, 均高于被动行走. 6 交互单元 Ape-X DPG 控制能力优于其他算法, 实现最高的 1 811 次稳定行走, 因此本文后续采用 6 交互单元 Ape-X DPG 进行分析. 从测试集中选择 2 组初始状态, 如表 3 所示.

表 3 机器人初始状态

Table 3 The initial states of the biped

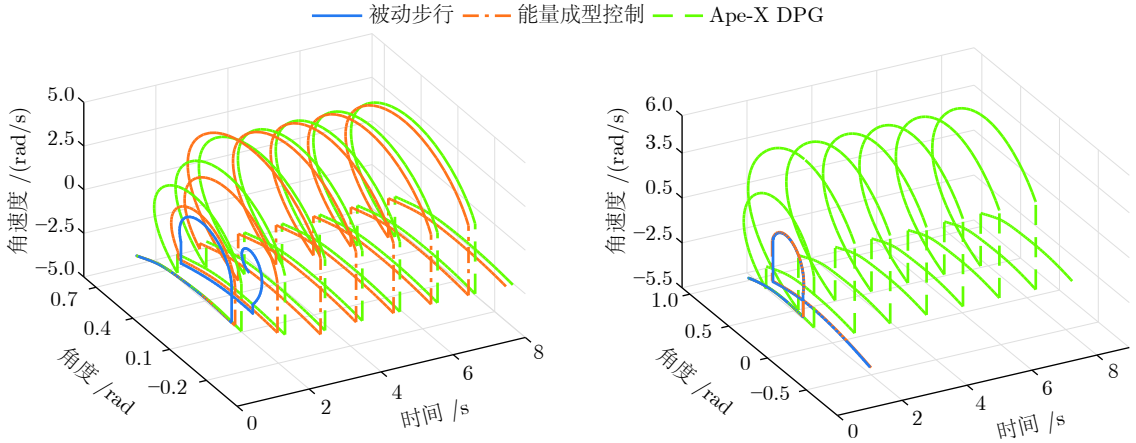
状态	θ_1 (rad)	$\dot{\theta}_1$ (rad/s)	$\dot{\theta}_2$ (rad/s)	ϕ
a	0.37149	−1.24226	2.97253	0.078
b	0.24678	−1.20521	0.15476	0.121

图 9 为以初始状态 a、b 行走的机器人前进方向左侧腿相空间示意图, 初始状态 a 时, 两种算法均可阻止机器人跌倒, 使机器人步态收敛至稳定状态, 并最终形成一致的运动轨迹. 相比于能量成型控制, Ape-X DPG 的控制过程更快, 仅通过两步的调整便使机器人步态趋近于稳定; 在初始状态 b 时, 能量成型控制失效, 无法抑制机器人的最终摔倒, 而 Ape-X DPG 依然可以完成机器人不稳定步态的快速恢复.

图 10 为初始状态 b 时两种算法的控制过程,

由于机器人初始状态的能量与不动点的能量接近, 使得能量成型控制效果微弱, 经过 3 步调整, 仍然无法抑制机器人跌倒. 而在 Ape-X DPG 作用下, 机器人第一步时 $\dot{\theta}_2$ 达到最高 2.842 rad/s, 高于能量成型控制时的 2.277 rad/s, 较大的 $\dot{\theta}_2$ 增大了机器人步幅、延长了第一步行走时间, 同时控制力矩的输入增加机器人系统机械能, 使碰撞后 $\dot{\theta}_1$、$\dot{\theta}_2$ 绝对值增大, 机器人状态则转移至 $s_2 = [0.39896, -1.60110, 1.38857, 0.121]$. 对比能量成型控制 $s_2 = [0.25749, -1.82706, -0.76343, 0.121]$可知, Ape-X DPG 在第一步时便使机器人步态向不动点 $x = [0.47213, -1.71598, 3.33070]$ 靠近. 此后, Ape-X DPG 作用力矩逐步减小并收束为 0, 最终机器人步态恢复稳定, 主动控制行走转为被动行走, 两种控制方法控制下的机器人棍状图如图 11 所示.

通过棍状图可直观看出, 在 Ape-X DPG 控制下机器人前 3 步中摆动腿摆动幅度逐步增大, 并在 4 步时开始稳定行走, 而在使用能量成型控制时, 机

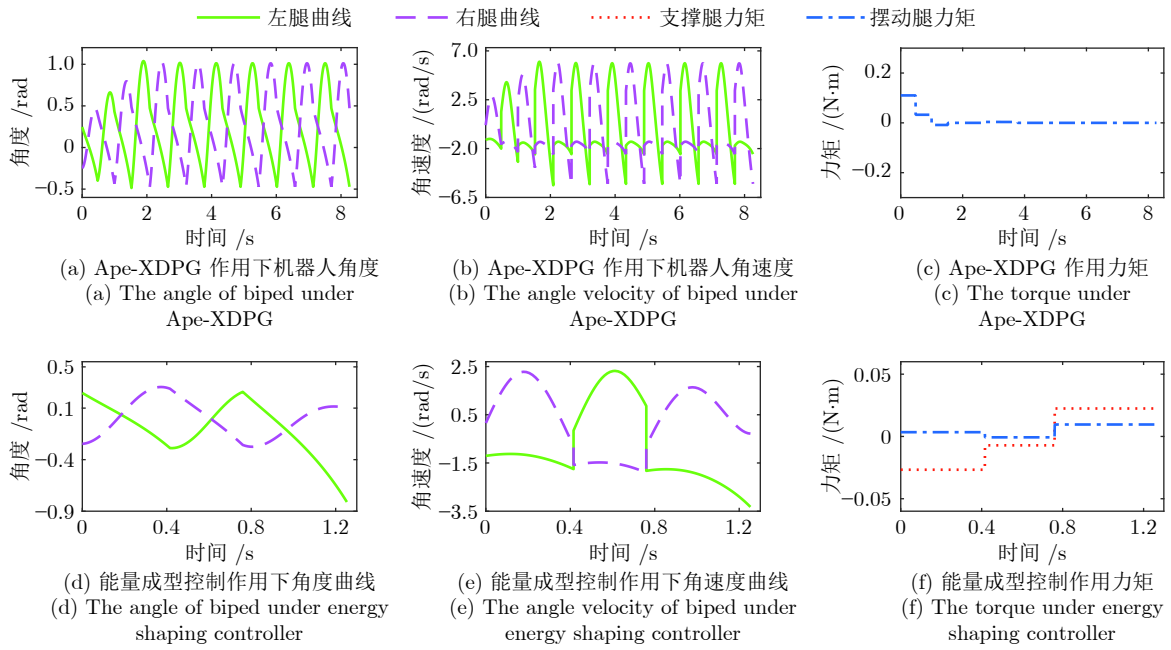


(a) 初始状态 a
(a) Initial state a

(b) 初始状态 b
(b) Initial state b

图 9　机器人左腿相空间图

Fig. 9　The phase plane of the right leg



(a) Ape-XDPG 作用下机器人角度
(a) The angle of biped under Ape-XDPG

(b) Ape-XDPG 作用下机器人角速度
(b) The angle velocity of biped under Ape-XDPG

(c) Ape-XDPG 作用力矩
(c) The torque under Ape-XDPG

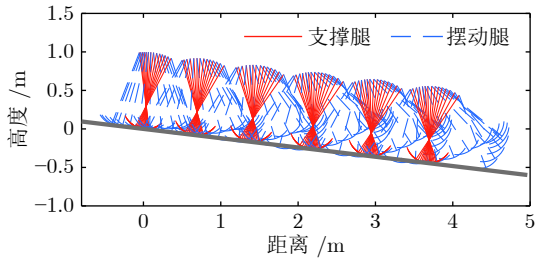
(d) 能量成型控制作用下角度曲线
(d) The angle of biped under energy shaping controller

(e) 能量成型控制作用下角速度曲线
(e) The angle velocity of biped under energy shaping controller

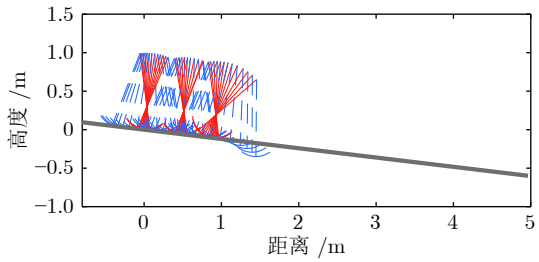
(f) 能量成型控制作用力矩
(f) The torque under energy shaping controller

图 10　初始状态 b 时机器人行走状态

Fig. 10　Biped walking state in initial state b



(a) Ape-XDPG 时
(a) Ape-XDPG



(b) 能量成型控制时
(b) Energy shaping controller

图 11 机器人行走过程棍状图
Fig. 11 The gait diagrams of the biped

器人在第 3 步时未能将摆动腿摆至地面上方, 进而引发了跌倒.

进一步的, 使用 Solidworks 建立机器人物理模型, 并基于 Matlab 中的 Simscape Multibody 进行物理仿真, 机器人模型参数同表 1, 模型如图 12 所示.

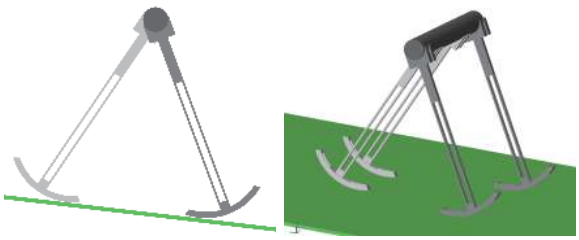
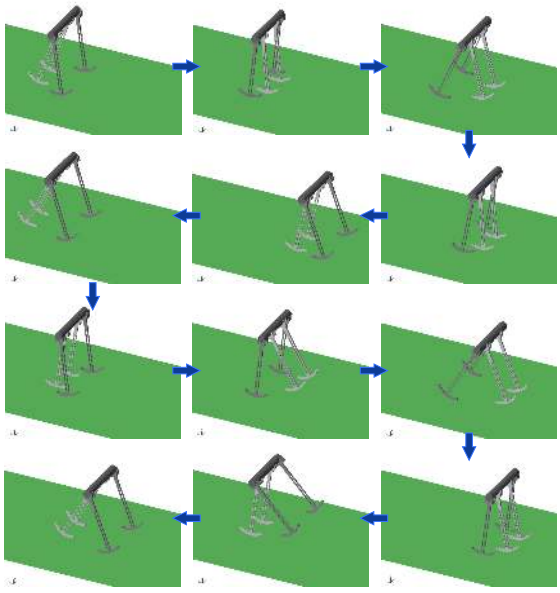


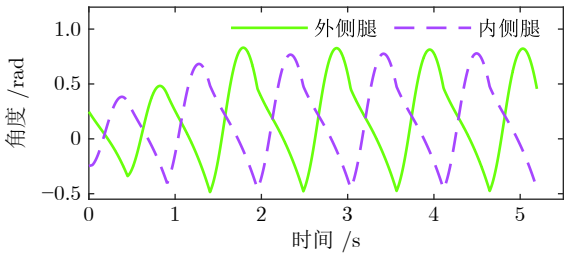
图 12 机器人物理模型示意图
Fig. 12 Sketch of the biped physical model

在机器人模型中, 外侧腿为灰色, 内侧腿为白色, 并定义外侧腿为物理仿真时的起始支撑腿. 为解决摆动阶段中摆动腿的擦地现象, 将摆动腿前摆时的足部碰撞参数置为 0, 回摆时恢复碰撞参数, 以实现足地碰撞. 在上述条件下, 初始状态 b 的机器人在 Ape-X DPG 控制下的前 10 步行走过程如图 13 所示.

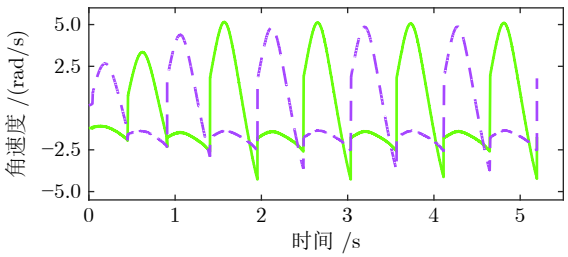
图 13 (a) 为机器人前 4 步的行走过程, 在图 13 (c) 中 0 s 时的角速度阶跃为机器人从空中释放后落地瞬间造成的速度突变. 从图 13 (b)、13 (c) 与图 10 (a)、10 (b) 对比可以看出, 物理仿真与数值仿



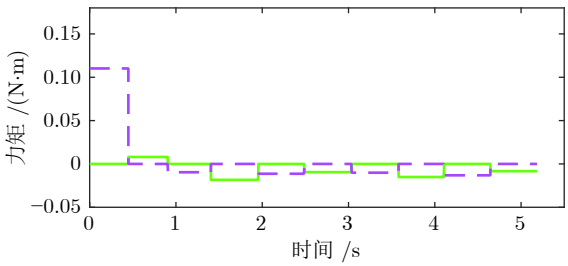
(a) 行走过程
(a) The gait diagrams



(b) 机器人角度
(b) The angle of biped



(c) 机器人角速度
(c) The angle velocity of biped



(d) 作用力矩
(d) The torque

图 13 机器人物理仿真
Fig. 13 Robot physics simulation

真在角度变化上大致相同, 但在角速度变化中存在

差异, 主要原因为物理仿真下机器人碰撞阶段无法实现完全非弹性碰撞. 因碰撞阶段的不同, 导致图 13 (d) 中 Ape-X DPG 作用力矩无法数值仿真时一样收敛至 0, 但依然可以阻止机器人的跌倒, 并使机器人在状态 $x = [0.46, -1.73, 1.81]$ 附近进行行走.

4.4　全局稳定性分析

为进一步刻画 Ape-X DPG 的控制能力, 采用胞映射法分别获得被动步行、能量成型控制、Ape-X DPG 三种情况下的机器人步态收敛域 (Basion of attraction, BOA). BOA 是机器人可稳定行走的初始状态集合, 其范围越大时机器人步行稳定性越高[40]. 为检测算法在不同坡度下的控制能力, 将 $\phi = [0.04, 0.15]$ 以 0.01 等间隔划分, 获得 12 组步行环境. 进一步将初始状态 S_1 中的 $x = [\theta_1, \dot{\theta}_1, \dot{\theta}_2]$ 划分为 6.4×10^4 个胞.

三种情况下机器人稳定行走胞数如图 14 所示, 在选取的 12 组坡度中, Ape-X DPG 控制下 BOA 稳定行走胞数均远高于被动步行与能量成型控制, 并在 $\phi = 0.1$ 时获得最大胞数为 55 649, 此时 BOA 可覆盖胞空间的 86.95 %, 而此时被动步行与能量成型控制胞数分别为 27 371、38 692, 覆盖胞空间为 42.76 %、58.89 %. 取 $\phi = 0.1$ 时的 BOA 进行绘制, 如图 15 所示, 其中绿色部分为 BOA 区域, 蓝色为跌倒步态区域. 图 15 (c) 中, Ape-X DPG 的 BOA 范围显著增加, 进一步证明 Ape-X DPG 可有效提高机器人步行稳定性.

5　总结

本文提出了一种稳定、高效的准被动双足机器人斜坡步态控制方法, 实现了 [0.04, 0.15] rad 斜坡范围内的机器人步态稳定控制. 在 DDPG 的基础上, 融合 PER 机制与 DPER 结构建立了 Ape-X DPG 分布式学习算法, 以加快样本采集速度、提高样本利用率、缩短学习时间. 将机器人视为智能体, 结合 Ape-X DPG 分布式的交互过程, 基于机器人行走特性的准确描述, 设计了 Episode 过程与奖励函数. 经学习后, Ape-X DPG 能够控制机器人在 2 000 组测试中完成 1 811 次稳定行走, 并在 $\phi = 0.1$ 时使机器人 BOA 覆盖 86.95 % 的胞空间. 相较于能量成型控制, Ape-X DPG 能够更有效地调节准被动

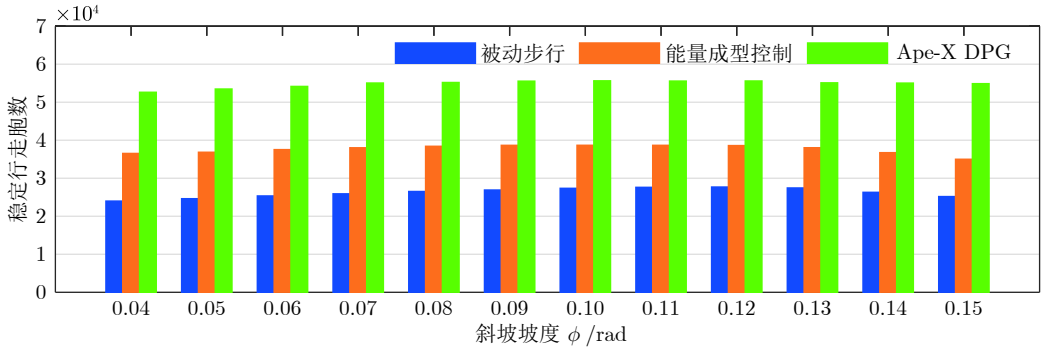
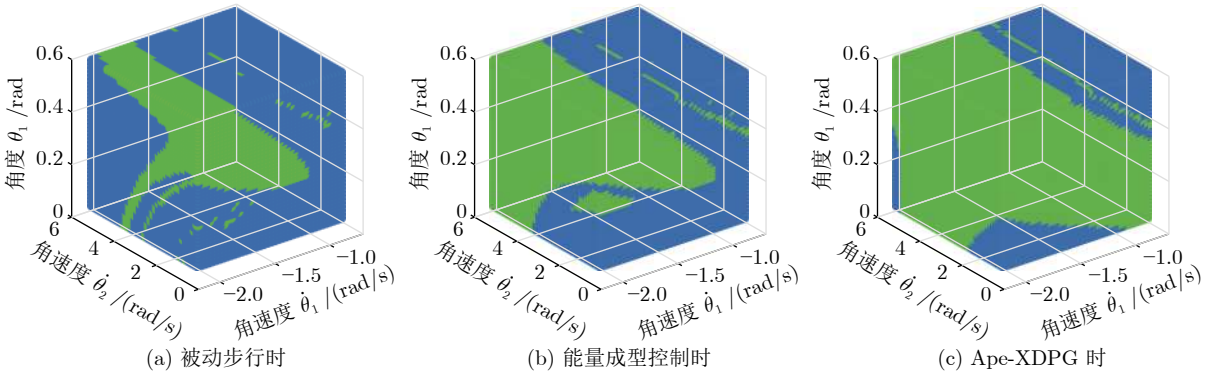


图 14　稳定行走胞数

Fig. 14　The number of the state walking



(a) 被动步行时 (a) Passive dynamic walking　(b) 能量成型控制时 (b) Energy shaping controller　(c) Ape-XDPG 时 (c) Ape-XDPG

图 15　$\phi = 0.1$ 时机器人步态收敛域

Fig. 15　The biped BOA when $\phi = 0.1$

双足机器人不稳定步态、抑制跌倒, 达到提高准被动双足机器人斜坡步行稳定性的目标.

References

1 Tian Yan-Tao, Sun Zhong-Bo, Li Hong-Yang, Wang Jing. A review of optimal and control strategies for dynamic walking bipedal robots. *Acta Automatica Sinica*, 2016, **42**(8): 1142−1157
(田彦涛, 孙中波, 李宏扬, 王静. 动态双足机器人的控制与优化研究进展. 自动化学报, 2016, **42**(8): 1142−1157)

2 Chin C S, Lin W P. Robust genetic algorithm and fuzzy inference mechanism embedded in a sliding-mode controller for an uncertain underwater robot. *IEEE/ASME Transactions on Mechatronics*, 2018, **23**(2): 655−666

3 Wang Y, Wang S, Wei Q P, Tan M, Zhou C, Yu J Z. Development of an underwater manipulator and its free-floating autonomous operation. *IEEE/ASME Transactions on Mechatronics*, 2016, **21**(2): 815−824

4 Wang Y, Wang S, Tan M, Zhou C, Wei Q P. Real-time dynamic Dubins-Helix method for 3-D trajectory smoothing. *IEEE Transactions on Control Systems Technology*, 2015, **23**(2): 730−736

5 Wang Y, Wang S, Tan M. Path generation of autonomous approach to a moving ship for unmanned vehicles. *IEEE Transactions on Industrial Electronics*, 2015, **62**(9): 5619−5629

6 Ma K Y, Chirarattananon P, Wood R J. Design and fabrication of an insect-scale flying robot for control autonomy. In: Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Hamburg, Germany: IEEE, 2015. 1558−1564

7 McGeer T. Passive dynamic walking. *The International Journal of Robotics Research*, 1990, **9**(2): 62−82

8 Bhounsule P A, Cortell J, Ruina A. Design and control of Ranger: An energy-efficient, dynamic walking robot. In: Proceedings of the 15th International Conference on Climbing and Walking Robots and the Support Technologies for Mobile Machines. Baltimore, MD, USA, 2012. 441−448

9 Kurz M J, Stergiou N. An artificial neural network that utilizes hip joint actuations to control bifurcations and chaos in a passive dynamic bipedal walking model. *Biological Cybernetics*, 2005, **93**(3): 213−221

10 Sun C Y, He W, Ge W L, Chang C. Adaptive neural network control of biped robots. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2017, **47**(2): 315−326

11 Sugimoto Y, Osuka K. Walking control of quasi passive dynamic walking robot "Quartet III" based on continuous delayed feedback control. In: Proceedings of the 2004 IEEE International Conference on Robotics and Biomimetics. Shenyang, China: IEEE, 2004. 606−611

12 Liu De-Jun, Tian Yan-Tao, Zhang Lei. Energy shaping control of under-actuated biped robot. *Journal of Mechanical Engineering*, 2012, **48**(23): 16−22
(刘德君, 田彦涛, 张雷. 双足欠驱动机器人能量成型控制. 机械工程学报, 2012, **48**(23): 16−22)

13 Spong M W, Holm J K, Lee D. Passivity-based control of bipedal locomotion. *IEEE Robotics & Automation Magazine*, 2007, **14**(2): 30−40

14 Liu Nai-Jun, Lu Tao, Cai Ying-Hao, Wang Shuo. A review of robot manipulation skills learning methods. *Acta Automatica Sinica*, 2019, **45**(3): 458−470
(刘乃军, 鲁涛, 蔡莹皓, 王硕. 机器人操作技能学习方法综述. 自动化学报, 2019, **45**(3): 458−470)

15 Tedrake R, Zhang T W, Seung H S. Stochastic policy gradient reinforcement learning on a simple 3D biped. In: Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Sendai, Japan: IEEE, 2004. 2849−2854

16 Hitomi K, Shibata T, Nakamura Y, Ishii S. Reinforcement learning for quasi-passive dynamic walking of an unstable biped robot. *Robotics and Autonomous Systems*, 2006, **54**(12): 982−988

17 Ueno T, Nakamura Y, Takuma T, Shibata T, Hosoda K, Ishii S. Fast and stable learning of quasi-passive dynamic walking by an unstable biped robot based on off-policy natural actor-critic. In: Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems. Beijing, China: IEEE, 2006. 5226−5231

18 Liu Quan, Zhai Jian-Wei, Zhang Zong-Zhang, Zhong Shan, Zhou Qian, et al. A survey on deep reinforcement learning. *Chinese Journal of Computers*, 2018, **41**(1): 1−27
(刘全, 翟建伟, 章宗长, 钟珊, 周倩, 章鹏, 等. 深度强化学习综述. 计算机学报, 2018, **41**(1): 1−27)

19 Kendall A, Hawke J, Janz D, Mazur P, Reda D, Allen J M, et al. Learning to drive in a day [Online], available: https://arxiv.org/abs/1807.00412, July 1, 2018

20 Wang Yun-Peng, Guo Ge. Signal priority control for trams using deep reinforcement learning. *Acta Automatica Sinica*, 2019, **45**(12): 2366−2377
(王云鹏, 郭戈. 基于深度强化学习的有轨电车信号优先控制. 自动化学报, 2019, **45**(12): 2366−2377)

21 Zhang Yi-Ke, Zhang Peng-Yuan, Yan Yong-Hong. Data augmentation for language models via adversarial training. *Acta Automatica Sinica*, 2018, **44**(5): 891−900
(张一珂, 张鹏远, 颜永红. 基于对抗训练策略的语言模型数据增强技术. 自动化学报, 2018, **44**(5): 891−900)

22 Andreas J, Rohrbach M, Darrell T, Klein D. Learning to compose neural networks for question answering. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. San Diego, California, USA: Association for Computational Linguistics, 2016. 1545−1554

23 Zhang X X, Lapata M. Sentence simplification with deep reinforcement learning. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. Copenhagen, Denmark: Association for Computational Linguistics, 2017. 584−594

24 Zhao Yu-Ting, Han Bao-Ling, Luo Qing-Sheng. Walking stability control method based on deep Q-network for biped robot on uneven ground. *Journal of Computer Applications*, 2018, **38**(9): 2459−2463
(赵玉婷, 韩宝玲, 罗庆生. 基于deep Q-network双足机器人非平整地面行走稳定性控制方法. 计算机应用, 2018, **38**(9): 2459−2463)

25 Mnih V, Kavukcuoglu K, Silver D, Rusu A A, Veness J, Bellemare M G, et al. Human-level control through deep reinforcement learning. *Nature*, 2015, **518**(7540): 529−533

26 Kumar A, Paul N, Omkar S N. Bipedal walking robot using deep deterministic policy gradient. In: Proceedings of the 2018 IEEE Symposium Series on Computational Intelligence. Bengaluru, India: IEEE, 2018.

27 Lillicrap T P, Hunt J J, Pritzel A, Heess N, Erez T, Tassa Y, et al. Continuous control with deep reinforcement learning [Online], available: https://arxiv.org/abs/1509.02971, September 9, 2015

28 Song D R, Yang C Y, McGreavy C, Li Z B. Recurrent deterministic policy gradient method for bipedal locomotion on rough terrain challenge. In: Proceedings of the 15th International Conference on Control, Automation, Robotics and Vision. Singapore: IEEE, 2018. 311−318

29 Todorov E, Erez T, Tassa Y. MuJoCo: A physics engine for model-based control. In: Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems. Vilamoura-Algarve, Portugal: IEEE. 2012. 5026−5033

30 Palanisamy P. Hands-On intelligent agents with openai gym: Your guide to developing AI agents using deep reinforcement learning. Birmingham, UK: Packt Publishing Ltd., 2018.

31 Schaul T, Quan J, Antonoglou I, Silver D. Prioritized experi-

ence replay. In: Proceedings of the International Conference on Learning Representations 2016. San Juan, Puerto Rico, 2016. 322−355

32 Horgan D, Quan J, Budden D, Barth-Maron G, Hessel M, van Hasselt H, et al. Distributed prioritized experience replay. In: Proceedings of the International Conference on Learning Representations 2018. Vancouver, Canada, 2018.

33 Zhao J, Wu X G, Zang X Z, Yang J H. Analysis of period doubling bifurcation and chaos mirror of biped passive dynamic robot gait. *Chinese Science Bulletin*, 2012, **57**(14): 1743−1750

34 Silver D, Lever G, Heess N, Degris T, Wierstra D, Riedmiller M. Deterministic policy gradient algorithms. In: Proceedings of the 31st International Conference on International Conference on Machine Learning. Beijing, China, 2014. I-387−I-395

35 Sutton R S, Barto A G. Reinforcement Learning: An Introduction. Cambridge: MIT Press, 1998.

36 Zhao J, Wu X G, Zhu Y H, Li G. The improved passive dynamic model with high stability. In: Proceedings of the 2009 International Conference on Mechatronics and Automation. Changchun, China: IEEE, 2009. 4687−4692

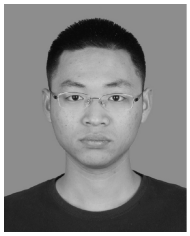
37 Abadi M, Barham P, Chen J M, Chen Z F, Davis A, Dean J, et al. TensorFlow: A system for large-scale machine learning. In: Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation. Savannah, USA: USENIX Association, 2016. 265−283

38 Kingma D P, Ba J. Adam: A method for stochastic optimization. In: Proceedings of the 3rd International Conference for Learning Representations. San Diego, USA, 2015.

39 Plappert M, Houthooft R, Dhariwal P, Sidor S, Chen R Y, Chen X, et al. Parameter space noise for exploration [Online], available: https://arxiv.org/abs/1706.01905, June 6, 2017

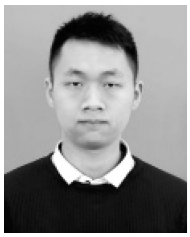
40 Schwab A L, Wisse M. Basin of attraction of the simplest walking model. In: Proceedings of the ASME 2001 Design Engineering Technical Conferences and Computers and Information in Engineering Conference. Pittsburgh, Pennsylvania: ASME, 2001. 531−539

吴晓光 燕山大学电气工程学院副教授, 2012 年获得哈尔滨工业大学博士学位. 主要研究方向为双足机器人, 三维虚拟视觉重构.
E-mail: wuxiaoguang@ysu.edu.cn
(**WU Xiao-Guang** Associate professor at the School of Electrical Engineering, Yanshan University. He received his Ph. D. degree in 2012 from Harbin University of Technology. His research interest covers biped robot and 3D virtual vision reconstruction.)

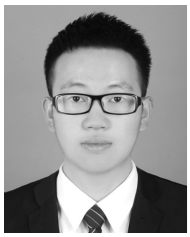


刘绍维 燕山大学电气工程学院硕士研究生. 主要研究方向为深度强化学习, 双足机器人. 本文通信作者.
E-mail: lwsalpha@outlook.com
(**LIU Shao-Wei** Master student at the School of Electrical Engineering, Yanshan University. His research interest covers deep reinforcement learning, biped robot. Corresponding author of this paper.)

杨 磊 燕山大学电气工程学院硕士研究生. 主要研究方向为双足机器人稳定性分析.
E-mail: 15733513567@163.com
(**YANG Lei** Master student at the School of Electrical Engineering, Yanshan University. His main research interest is stability analysis of biped robot.)



邓文强 燕山大学电气工程学院硕士研究生. 主要研究方向为生成对抗网络, 人体运动协调性分析.
E-mail: dengwq24@163.com
(**DENG Wen-Qiang** Master student at the School of Electrical Engineering, Yanshan University. His research interest covers generating confrontation networks and the analysis of human motion coordination.)



贾哲恒 燕山大学电气工程学院硕士研究生. 主要研究方向为人体姿态估计, 目标识别, 深度学习.
E-mail: jiazheheng@163.com
(**JIA Zhe-Heng** Master student at the School of Electrical Engineering, Yanshan University. His research interest covers human pose estimation, object detection, and deep learning.)