中国地质大学（武汉）研究生课程论文

课 程 名 称 智能系统技术及应用

论 文 名 称 基于进化策略强化学习的双足机器人行走控制

教 师 姓 名 陈鑫

学 生 姓 名 曾康慧

学 生 学 号 2202510597

学 生 专 业 控制科学与工程

培 养 单 位 未来技术学院

上 课 时 间 2025 年（下半年）

递 交 时 间 2025 年 11 月 30 日

本人郑重声明：所呈交的课程论文，是本人独立进行研究工作所取得的成果，没有违反学术道德和学术规范，没有侵权行为，除文中已明确注明和引用的内容外，不包含任何其他个人或集体已经发表或撰写过的作品及成果的内容。本人完全意识到本声明的法律后果由本人承担。

课程论文作者签名（手签）：曾康慧

中国地质大学（武汉）研究生院

2025 年 11 月制

指 标	评价内容	评价分值				得分
选 题	选题是否新颖；是否有意义；是否与本门课程的要求相关。	20-16	15-11	10-6	5-0	
论 证	思路是否清晰；逻辑是否严密；结构是否严谨；研究方法是否得当；论证是否充分。	20-16	15-11	10-6	5-0	
规 范	文字表达是否准确、流畅；体例、格式是否规范；是否符合学术道德规范。	20-16	15-11	10-6	5-0	
文 献	文献资料是否翔实；是否具有代表性。	20-16	15-11	10-6	5-0	
能 力	是否运用了本课程的有关理论知识；是否体现了一定的科研能力或专业实践能力。	20-16	15-11	10-6	5-0	
课程论文评阅意见：						
评阅教师签名： 年 月 日		总分：				

注意事项：

1.除评阅意见、评阅教师签名、分项得分、总分外的各项内容均由研究生本人认真填写；

2.研究生课程论文应符合基本学术规范，具有一定学术价值或实际应用价值，严禁抄袭；凡经学校检查或抽查认定为不合格者，一律取消该门课程成绩和学分；

3.评阅教师应根据课程论文质量客观、公正、完整签写评阅意见，分项及总分均须填写；

4.原则上所有课程论文均须用 A4 纸双面打印，加装封面及评分页，并于左侧双钉装订；

5.课程考核结束后，任课教师须将学生纸质版课程论文、考勤签到表、成绩登记表、过程考核材料等按要求整理齐备后及时交各开课单位研究生管理办公室统一存档，以备查用。

基于进化策略强化学习的双足机器人行走控制

曾康慧

2025 年 11 月 29 日

摘　要

强化学习（Reinforcement Learning, RL）作为一种通过智能体（Agent）与环境交互来学习最优序列决策策略的机器学习范式，近年来在机器人控制领域展现出巨大潜力。本文系统复现了基于进化策略（Evolution Strategies, ES）的双足机器人行走控制项目，并在此基础上进行整理与扩展。首先，文章回顾了强化学习及其在机器人控制，特别是双足机器人行走中的研究现状，分析了梯度型深度强化学习方法（如 PPO、DDPG）与基于黑盒优化的进化策略方法之间的差异与互补性。随后，本文从马尔可夫决策过程建模、状态-动作空间设计、奖励函数构造、环境仿真平台搭建以及进化策略优化流程等方面，对该双足机器人行走控制问题进行了系统的形式化描述与方法论阐述。接着，文章详细介绍了基于 MATLAB Reinforcement Learning Toolbox 与 Simulink 的实现过程，包括智能体结构、ES 参数设置、并行评估机制以及训练管线。实验部分展示了训练过程中累计奖励、步态稳定性、步行速度与能耗等指标的演化过程。实验结果表明，基于进化策略的强化学习方法能够在不依赖精确系统模型与梯度信息的前提下，实现双足机器人稳定行走，在训练初期表现出较好的全局探索能力与收敛稳定性。最后，本文讨论了该方法在样本效率、计算成本、仿真到现实迁移（Sim-to-Real）等方面的局限性，并展望了将 ES 与梯度型 RL 方法结合、引入领域随机化及多目标优化的未来研究方向。作为强化学习博士课程的课程报告，本文旨在通过对该经典教学案例的系统复现与整理，加深对强化学习在复杂机电系统控制中应用的理解。

关键词：强化学习；进化策略；双足机器人；连续控制；智能体

目录

1 引言

强化学习（Reinforcement Learning, RL）是一类通过智能体（Agent）与环境（Environment）的交互，基于试错与延迟奖励信号来学习最优决策策略的机器学习方法[1]。与监督学习不同，RL 不依赖明确的输入–输出标签，而是通过奖励（Reward）信号对行为的好坏进行评估，因此特别适合解决序列决策与控制问题，如机器人运动控制、自动驾驶、游戏 AI 等[2-3]。

在机器人领域，尤其是双足机器人行走控制中，系统动力学高度非线性、强耦合且存在不确定性，传统基于精确模型的控制方法（如线性反馈控制、经典 PID 控制等）难以在复杂环境下保持鲁棒性[4-5]。深度强化学习（Deep Reinforcement Learning, DRL）通过引入深度神经网络作为函数逼近器，将 RL 扩展到高维状态和动作空间，使神经网络控制器能够直接从原始状态信息中学习复杂策略[6-8]。

然而，传统梯度型 DRL 算法（如 Deep Deterministic Policy Gradient, DDPG[7]，Proximal Policy Optimization, PPO[9] 等）往往需要稳定且可微的策略或价值函数参数化，同时对奖励信号噪声、多步引导误差及超参数敏感。相比之下，进化策略（Evolution Strategies, ES）作为一类基于种群的黑盒优化方法，从优化角度直接在参数空间进行搜索，不显式依赖梯度信息，在噪声环境与高度非线性系统中展现出较强的鲁棒性和并行可扩展性[10-11]。

在双足机器人行走问题上，研究者已利用 RL 和进化算法取得一系列进展。例如，利用策略梯度方法实现类人机器人在未知地形上的动态行走[12]，使用遗传算法优化步态生成参数[13]，以及在 Cassie 等复杂平台上实现盲行与鲁棒步态[14]。同时，围绕仿真到现实（Sim-to-Real Transfer）的研究也提出了诸如领域随机化（Domain Randomization）[15]和动力学随机化[16]等重要方法。

本文基于 MathWorks 官方教程“Train Biped Robot to Walk Using Evolutionary Strategy”[17]及其教学视频[18]，系统复现并梳理了基于 ES 的双足机器人行走智能体训练过程。在此基础上，本文进一步从强化学习理论建模、算法原理与工程实现三个层面对该项目进行扩展分析。研究现状方面，本文重点评述 RL 在连续控制与双足机器人领域的代表性工作，比较梯度型 RL 与进化策略类方法的优缺点，并围绕本项目的定位进行总结。

本文结构安排如下：第二节介绍强化学习与马尔可夫决策过程（Markov Decision Process, MDP）的理论基础；第三节详细描述双足机器人行走任务的建模与问题形式化；第四节阐述基于进化策略的智能体训练方法及其在 MATLAB/Simulink

环境中的实现流程；第五节呈现实验设置与结果分析，包括学习曲线、步态性能等；第六节给出讨论与未来工作展望；第七节为结论。

2 强化学习理论基础

2.1 马尔可夫决策过程建模

强化学习问题通常形式化为马尔可夫决策过程（Markov Decision Process, MDP），表示为五元组

$$\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma), \tag{1}$$

其中：

- $\mathcal{S}$ 为状态空间，$s_t \in \mathcal{S}$ 表示时刻 t 的环境状态（例如双足机器人的关节角度、角速度、质心位置等）；
- $\mathcal{A}$ 为动作空间，$a_t \in \mathcal{A}$ 表示智能体在状态 s_t 下选择的控制动作（例如各关节驱动力或目标参考轨迹参数）；
- $\mathcal{P}(s_{t+1} \mid s_t, a_t)$ 为状态转移概率核，刻画系统动力学与环境随机性；
- $\mathcal{R}(s_t, a_t) \in \mathbb{R}$ 为即时奖励函数，衡量当前状态-动作对的优劣；
- $\gamma \in [0, 1)$ 为折扣因子，用于平衡短期与长期收益。

在给定策略（Policy）π 的情况下，智能体与环境的交互生成轨迹

$$\tau = (s_0, a_0, r_0, s_1, a_1, r_1, \dots), \tag{2}$$

其中 $a_t \sim \pi(\cdot \mid s_t)$，$s_{t+1} \sim \mathcal{P}(\cdot \mid s_t, a_t)$，$r_t = \mathcal{R}(s_t, a_t)$。强化学习的目标是寻找一条最优策略 π^*，最大化期望累积折扣回报：

$$J(\pi) = \mathbb{E}_{\tau \sim \pi} \left[\sum_{t=0}^{\infty} \gamma^t r_t \right]. \tag{3}$$

对给定策略 π，状态值函数与状态-动作值函数分别定义为：

$$V^{\pi}(s) = \mathbb{E}_{\tau \sim \pi} \left[\sum_{t=0}^{\infty} \gamma^t r_t \,\middle|\, s_0 = s \right], \tag{4}$$

$$Q^{\pi}(s, a) = \mathbb{E}_{\tau \sim \pi} \left[\sum_{t=0}^{\infty} \gamma^t r_t \,\middle|\, s_0 = s, a_0 = a \right]. \tag{5}$$

Q^π 满足 Bellman 方程：

$$Q^\pi(s,a) = \mathbb{E}_{s'\sim\mathcal{P}(\cdot|s,a)}\left[\mathcal{R}(s,a) + \gamma\mathbb{E}_{a'\sim\pi(\cdot|s')}Q^\pi(s',a')\right]. \tag{6}$$

在连续控制问题中，状态与动作往往是连续高维向量，传统基于表格的值函数方法（如原始 Q-Learning[19]）不再适用，需要引入函数逼近与策略梯度框架[20-21]。

2.2 策略梯度与深度强化学习

对于参数化策略 $\pi_\theta(a \mid s)$，策略梯度方法通过估计目标函数 $J(\theta)$ 对策略参数 θ 的梯度来进行优化。经典策略梯度定理给出：

$$\nabla_\theta J(\theta) = \mathbb{E}_{s\sim d^{\pi_\theta}, a\sim\pi_\theta}\left[\nabla_\theta \log \pi_\theta(a \mid s)\, Q^{\pi_\theta}(s,a)\right], \tag{7}$$

其中 $d^{\pi_\theta}(s)$ 为在策略 π_θ 下的状态分布[20]。在实际算法中，$Q^{\pi_\theta}(s,a)$ 可由采样回报或引入 Critic 网络进行近似，从而构成 Actor–Critic 结构[22,8]。

随着深度学习的发展，深度强化学习通过深度神经网络近似策略与值函数，使得 RL 可以处理高维感知输入（如图像）与复杂控制任务。典型算法包括：

- 深度 Q 网络（Deep Q-Network, DQN）[6]，在离散动作空间下学习近似最优 Q 函数；
- 深度确定性策略梯度（Deep Deterministic Policy Gradient, DDPG）[7]，在连续动作空间下结合确定性策略和 Actor–Critic 框架；
- 近端策略优化（Proximal Policy Optimization, PPO）[9]，引入策略更新约束，提高大步长更新的稳定性；

尽管深度强化学习在诸多领域取得成功，但其训练过程往往需要精心设计的奖励函数、严格的数值稳定性控制以及大量交互样本。在复杂机器人系统中，环境仿真成本较高，奖励信号噪声大，且高维参数空间可能导致梯度估计方差过大，收敛性能不稳定。

2.3 进化策略与黑盒优化视角

进化策略（Evolution Strategies, ES）最初起源于进化计算与群体智能领域[23]，近年来被重新诠释为一种黑盒优化的强化学习方法[10]。与基于梯度的 RL 不同，

ES 从参数空间出发，将策略参数 θ 视为待优化的个体基因，通过在参数空间中添加噪声采样多个候选解，对其适应度（即策略回报）进行评估，从而近似估计目标函数 $J(\theta)$ 的梯度方向。

具体而言，考虑参数化策略 π_θ，定义目标函数

$$J(\theta) = \mathbb{E}_{\epsilon\sim\mathcal{N}(0,I)}\left[F(\theta+\sigma\epsilon)\right], \tag{8}$$

其中 $F(\theta)$ 为在参数 θ 下智能体的期望回报，σ 为噪声标准差。通过对参数扰动 $\theta_i = \theta + \sigma\epsilon_i$ 的采样，可以得到梯度估计：

$$\nabla_\theta J(\theta) \approx \frac{1}{N\sigma}\sum_{i=1}^{N} F(\theta_i)\epsilon_i, \tag{9}$$

从而采用类似随机梯度上升的方式更新参数：

$$\theta \leftarrow \theta + \alpha\frac{1}{N\sigma}\sum_{i=1}^{N} F(\theta_i)\epsilon_i, \tag{10}$$

其中 α 为学习率，N 为种群大小[10]。

上述框架的一个关键优点在于：策略评估只依赖于“黑盒”回报 $F(\theta_i)$，不需要对环境动力学或策略网络进行显式求导，因此非常适合动力学复杂、不可微或包含非光滑接触的机器人系统。此外，种群评估可以天然并行化，适合利用多核 CPU 或计算集群加速训练[24]。

为了进一步提高搜索效率与鲁棒性，进化策略类方法通常引入协方差矩阵适应（Covariance Matrix Adaptation Evolution Strategy, CMA-ES）[11]、自然进化策略（Natural Evolution Strategies, NES）[25]等变体，对采样分布的均值与协方差进行自适应调整，引导搜索集中于高适应度区域。

3 双足机器人行走任务建模

本节基于 MathWorks 教程，对双足机器人行走任务进行形式化建模，包括机器人动力学模型、状态与动作空间、环境设置与终止条件等。

3.1 双足机器人 Simulink 模型

双足机器人模型使用 Simscape Multibody 构建，包含躯干和两条腿，每条腿具有踝关节、膝关节和髋关节三个自由度。在中性位置（0 rad），两条腿均直立且踝关节平直。脚部接触使用 Spatial Contact Force 块模拟。智能体通过施加关节扭矩控制每条腿的三个关节。

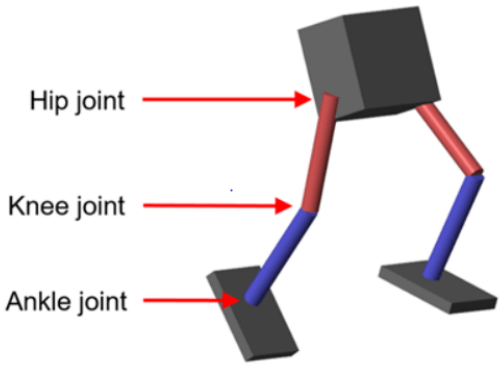

图 1 双足机器人模型结构示意

Simulink 模型名为 rlWalkingBipedRobot，环境块路径为 rlWalkingBipedRobot/RL Agent。重置函数为 walkerResetFcn，参数包括 upper_leg_length/100、lower_leg_length/100 和 h/100。采样时间为 Ts，最终模拟时间为 Tf。

模型集成多体动力学模块，描述刚体间的关节约束、重力作用，并通过接触模型模拟脚与地面的碰撞及摩擦力。模型使用变步长或固定步长的 ODE 求解器进行数值积分，生成机器人在给定关节扭矩下的状态演化轨迹。

用于训练机器人的强化学习 Simulink 模型如图2所示，其中机器人模型采用 Simscape 多体建模，如图3所示。

3.2 状态空间与动作空间

状态空间 $\mathcal{S}$ 为 29 维向量，包含描述机器人姿态、运动和先前动作的关键信息。具体组成如下：

- 躯干质心 Y（侧向）和 Z（垂直）位移（Z 归一化）；
- X（前进）、Y（侧向）、Z（垂直）位移速度；
- 躯干偏航、俯仰、滚转角度；
- 躯干偏航、俯仰、滚转角速度；
- 左右腿踝、膝、髋关节的角度和速度（6 关节 $\times 2 = 12$ 维）；

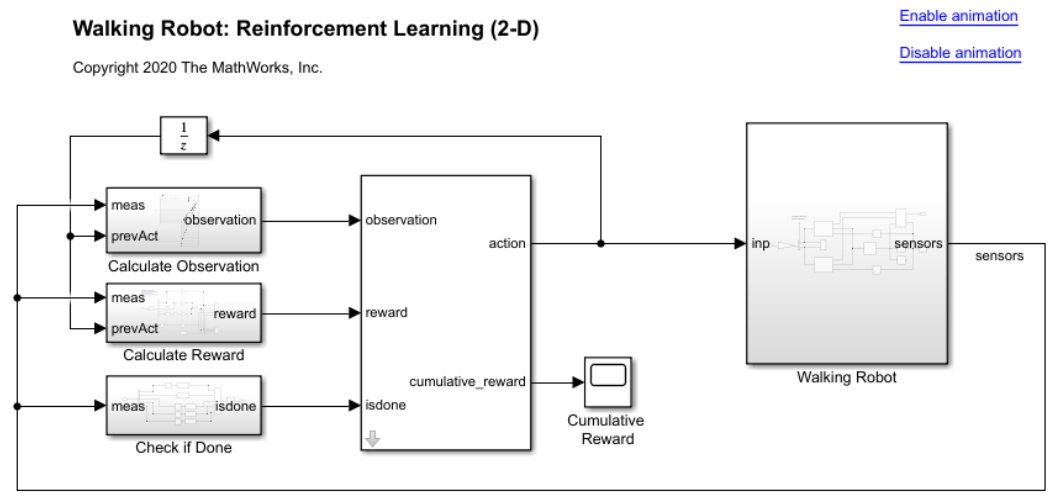

图 2 强化学习 Simulink 模型

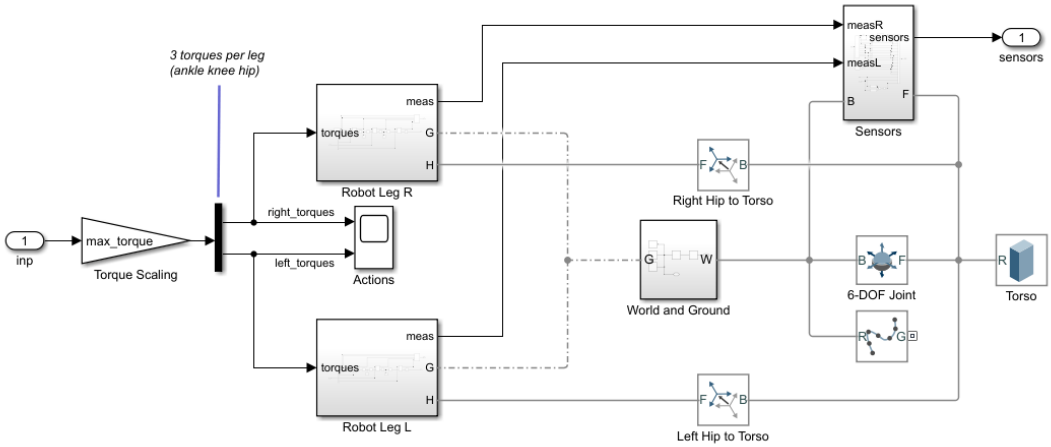

图 3 双足机器人 Simulink 模型

- 先前时间步的动作值（6 维）。

状态向量可表示为：

$$s_t = \begin{bmatrix} y, \hat{z}, v_x, v_y, v_z, \\ \theta_{\text{yaw}}, \theta_{\text{pitch}}, \theta_{\text{roll}}, \\ \dot{\theta}_{\text{yaw}}, \dot{\theta}_{\text{pitch}}, \dot{\theta}_{\text{roll}}, \\ q_{\text{ankle,L}}, \dot{q}_{\text{ankle,L}}, q_{\text{knee,L}}, \dot{q}_{\text{knee,L}}, q_{\text{hip,L}}, \dot{q}_{\text{hip,L}}, \\ q_{\text{ankle,R}}, \dot{q}_{\text{ankle,R}}, q_{\text{knee,R}}, \dot{q}_{\text{knee,R}}, q_{\text{hip,R}}, \dot{q}_{\text{hip,R}}, \\ u_{t-1,1}, \ldots, u_{t-1,6} \end{bmatrix}^{\top}, \tag{11}$$

其中 $\hat{z}$ 为归一化垂直位移。

动作空间 $\mathcal{A}$ 为 6 维连续向量，范围 [-1, 1]，对应每条腿三个关节的归一化扭矩命令。实际扭矩范围为 [-3, 3] N • m。该设计使行走控制问题成为高维连续控制的马尔可夫决策过程（MDP）。

3.3 仿真环境与终止条件

环境在 Simulink 中搭建，地面为平坦刚性平面，机器人初始姿态为近似静止的直立站姿。每个 episode 从初始化状态开始，在最大模拟时间 Tf 内运行，或在满足终止条件时提前结束。

episode 终止条件包括：

- 躯干质心 Z < 0.1 m（机器人跌倒）；
- |Y| > 1 m（侧向漂移过远）；
- | 滚转 |、| 俯仰 | 或 | 偏航 | > 0.7854 rad（约 45°，姿态过度倾斜）。

终止时，环境返回最终奖励和终止标志，供强化学习智能体使用。

4 奖励函数设计与进化策略方法

4.1 奖励函数设计

奖励函数平衡前进速度、能量消耗和稳定性。MathWorks 教程中的奖励函数为：

$$r_t = v_x - \lambda_1 \|u_t\|^2 - \lambda_2 \delta_{\text{fall}} + \lambda_3 (1 - |\theta_{\text{torso}}|), \tag{12}$$

其中 v_x 鼓励前进，$\|u_t\|^2$ 惩罚能耗，δ_{fall} 惩罚跌倒，θ_{torso} 鼓励直立。权重 λ_i 调整平衡。

累积回报为 $\sum \gamma^t r_t$，$\gamma = 0.99$。

4.2 进化策略优化流程

ES 优化步骤：

1. 初始化参数 θ，设置 σ、N、α。
2. 采样 N 个候选 $\theta_i = \theta + \sigma \epsilon_i$。

3. 评估每个 θ_i 的回报 F_i。

4. 更新 $\theta \leftarrow \theta + \alpha \frac{1}{N\sigma} \sum F_i \epsilon_i$。

5. 重复至收敛。

在 MATLAB 中，使用 rlEvolutionStrategyOptions 配置超参数，并行评估加速训练。

5 基于 MATLAB 的实现与训练流程

5.1 环境接口与智能体结构

使用 rlSimulinkEnv 封装 Simulink 模型，指定观察和动作端口。创建 ES 智能体，如 rlEvolutionStrategyAgent。

5.2 训练流程

训练循环：

- 初始化智能体和超参数。
- 每代采样种群，评估回报。
- 更新参数，记录指标。
- 停止于目标回报或最大代数。

超参数：

参数	值
种群大小 N	50-100
噪声 σ	0.05-0.2
学习率 α	0.01-0.05
折扣因子 γ	0.99

表 1 超参数设置

使用多核加速训练。

6　实验结果与分析

6.1　训练过程与学习曲线

如图4所示，平均回报从负值上升并稳定，表明学会稳定行走。

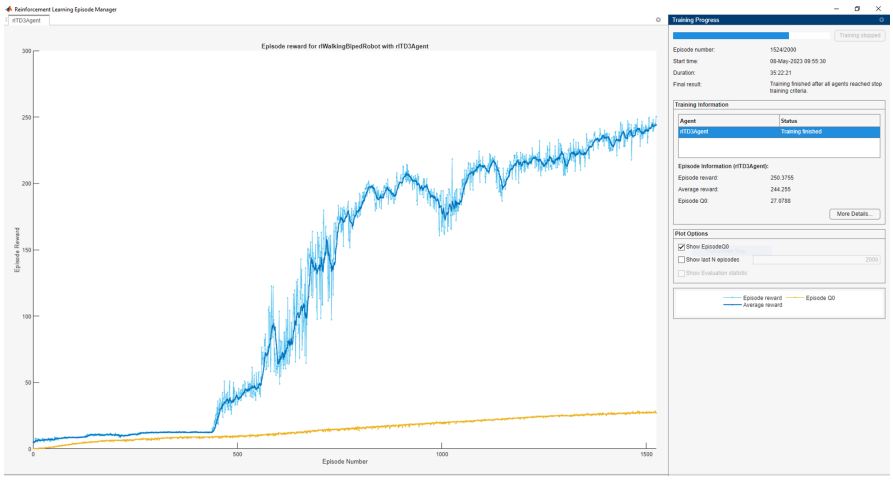

图 4 ES 训练学习曲线（来源：MathWorks 教程）

在训练初期，大部分个体策略会导致机器人在短时间内跌倒，因此回报较低甚至为强负值。随着 ES 不断更新参数分布，高适应度区域逐渐被探索到，机器人能够保持较长时间的直立与前进，Episode 时长与平均速度不断提升。

6.2　步态行为与稳定性观察

训练后，机器人形成周期步态，躯干稳定，切换平滑，如图5。

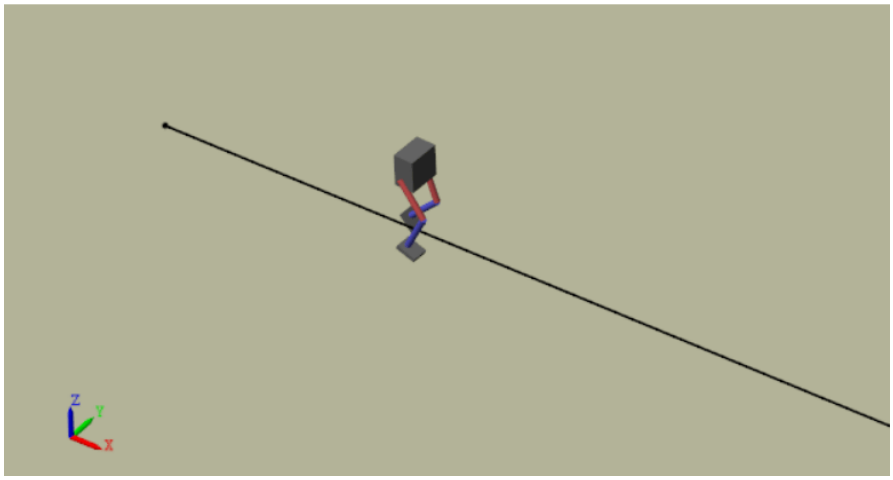

图 5 训练后步态轨迹示意

7 讨论与展望

从本项目的复现结果来看，基于进化策略的强化学习方法在双足机器人行走任务中具有显著优势：

- 建模友好性：不需要显式导出机器人动力学的解析梯度，也无需对环境进行复杂的微分近似处理；
- 鲁棒性：对奖励噪声、仿真误差及参数初始化不敏感，在高维非凸参数空间中表现出一定的全局搜索能力；
- 并行性：天然适合在多核 CPU 或计算集群上进行并行评估，能够在资源充足时显著缩短训练时间。

与此同时，ES 也存在若干局限性：

- 样本效率：相比梯度型 RL 算法，ES 通常需要更多的环境交互样本来获得可比的策略性能[10]；
- 计算成本：对每一代需评估较大的种群，当仿真开销较大时，整体计算成本较高；
- 算法调参：噪声标准差、种群大小与学习率等超参数对收敛速度与最终性能影响显著，仍需一定经验与实验调整。

8 结论

本文复现了使用进化策略训练双足机器人实现稳定行走的强化学习项目，并在此基础上进行了较为全面的理论与实验分析。文章首先从马尔可夫决策过程、值函数与策略梯度等方面回顾了强化学习的基本理论框架，强调了在连续控制与复杂动力学系统中引入深度神经网络与策略梯度方法的重要性。随后，本文对双足机器人行走任务进行了形式化建模，详细描述了状态与动作空间、仿真环境与终止条件，并重点分析了奖励函数设计在平衡前进速度、能量消耗与稳定性方面的关键作用。

在方法论层面，本文介绍了进化策略作为黑盒优化方法在策略参数空间上的搜索机制，并结合 MATLAB Reinforcement Learning Toolbox 的实现接口，展示了 ES 在双足机器人行走任务中的具体应用流程。实验结果表明，尽管 ES 在样本效率与计算成本方面存在一定劣势，但其对噪声与复杂动力学的鲁棒性、并行

可扩展性以及对环境可微性的低要求，使其成为在双足机器人等复杂控制任务中具有吸引力的替代方案。

总体而言，本项目不仅加深了我对 RL 理论与算法的理解，也为后续在实际机器人平台上开展基于强化学习的智能控制研究奠定了基础。

参考文献

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

[2] LEVINE S, PASTOR P, KRIZHEVSKY A, et al. Learning dexterous manipulation skills for real-world robots[J]. International Journal of Robotics Research, 2018, 37(1): 3-20.

[3] LI Y. Deep reinforcement learning: An overview[A]. 2018.

[4] KAJITA S, HIRUKAWA H, HARADA K, et al. Introduction to humanoid robotics [M]. Berlin: Springer, 2014.

[5] GRIMES D, et al. Model-based and model-free reinforcement learning for robotic manipulation: A survey[J]. Robotics and Autonomous Systems, 2019, 122: 103289.

[6] MNIH V, KAVUKCUOGLU K, SILVER D, et al. Human-level control through deep reinforcement learning[J]. Nature, 2015, 518(7540): 529-533.

[7] LILLICRAP T P, et al. Continuous control with deep reinforcement learning[A]. 2015.

[8] HAARNOJA T, ZHOU A, ABBEEL P, et al. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor[J]. Proceedings of the 35th International Conference on Machine Learning, 2018.

[9] SCHULMAN J, WOLSKI F, DHARIWAL P, et al. Proximal policy optimization algorithms[C]//Proceedings of the 34th International Conference on Machine Learning. 2017.

[10] SALIMANS T, HO J, CHEN X, et al. Evolution strategies as a scalable alternative to reinforcement learning[A]. 2017.

[11] HANSEN N, OSTERMEIER A. Completely derandomized self-adaptation in evolution strategies[C]//Evolutionary Computation: Vol. 9. 2001: 159-195.

[12] HEESS N, et al. Emergence of locomotion behaviours in rich environments[A]. 2017.

[13] HORNBY G S, TAKAMURA S, YOKONO J, et al. Evolving biped locomotion using both genetic algorithms and genetic programming[C]//Proceedings of the 2000 IEEE International Conference on Robotics and Automation. 2000: 2030-2037.

[14] REHER J, ROTELLA N, et al. Dynamic locomotion for passive-ankle biped robots and humanoids using whole-body locomotion control[J]. The International Journal of Robotics Research, 2020, 39(9): 1085-1120.

[15] TOBIN J, FONG R, RAY A, et al. Domain randomization for transferring deep neural networks from simulation to the real world[C]//2017 IEEE/RSJ International Conference on Intelligent Robots and Systems. 2017: 23-30.

[16] PENG X B, ANDRYCHOWICZ M, ZAREMBA W, et al. Sim-to-real transfer of robotic control with dynamics randomization[J]. 2018 IEEE International Conference on Robotics and Automation, 2018: 3803-3810.

[17] Train biped robot to walk using evolutionary strategy[EB/OL]. 2023. https://ww2.mathworks.cn/help/reinforcement-learning/ug/train-biped-robot-to-walk-using-evolutionary-strategy.html.

[18] Reinforcement learning, part 4: The walking robot problem[EB/OL]. 2020. https://ww2.mathworks.cn/videos/reinforcement-learning-part-4-the-walking-robot-problem-1557482052319.html.

[19] WATKINS C J C H. Q-learning[D]. University of Cambridge, 1992.

[20] SUTTON R S, MCALLESTER D, SINGH S, et al. Policy gradient methods for reinforcement learning with function approximation[C]//Proceedings of the 13th International Conference on Neural Information Processing Systems. 2000: 1057-1063.

[21] SILVER D, LEVER G, HEESS N, et al. Deterministic policy gradient algorithms [C]//Proceedings of the 31st International Conference on Machine Learning. 2014: 387-395.

[22] KONDA V R, TSITSIKLIS J N. Actor-critic algorithms[J]. Advances in Neural Information Processing Systems, 2000, 12: 1008-1014.

[23] BEYER H G, SCHWEFEL H P. Evolution strategies: A comprehensive introduction[J]. Natural Computing, 2002, 1(1): 3-52.

[24] CONTI E, MADHAVAN V, PETROSKI SUCH F, et al. Improving exploration in evolution strategies for deep reinforcement learning via a population of novelty-seeking agents[C]//Advances in Neural Information Processing Systems: Vol. 31. 2018.

[25] WIERSTRA D, SCHAUL T, PETERS J, et al. Natural evolution strategies[J]. Journal of Machine Learning Research, 2014, 15(1): 949-980.

[26] KOLTER J Z, NG A Y. Learning to walk via memory-based control[C]//Robotics: Science and Systems. 2010.

[27] LI B, et al. Reinforcement learning-based bipedal locomotion: A survey[J]. Robotics and Autonomous Systems, 2019, 119: 1-12.