



(12) 发明专利申请

(10) 申请公布号 CN 117742322 A

(43) 申请公布日 2024. 03. 22

(21) 申请号 202311649978.6

(22) 申请日 2023.12.04

(71) 申请人 人工智能与数字经济广东省实验室
(广州)

地址 510330 广东省广州市海珠区新港东
路2429号首层自编051房

申请人 中山大学

(72) 发明人 黄凯 张子韬 赵子健 陶若怡

(74) 专利代理机构 广州粤高专利商标代理有限
公司 44102

专利代理师 王晓玲

(51) Int. Cl.

G05D 1/43 (2024.01)

G05D 1/622 (2024.01)

权利要求书4页 说明书10页 附图2页

(54) 发明名称

一种基于强化学习的小型仿生鼠机器人的
运动控制方法

(57) 摘要

本发明涉一种基于强化学习的小型仿生鼠机器人的运动控制方法。首先根据仿生鼠机器人的结构特点构建了基于最小化有效信息提取的信息处理流,定义了强化学习过程中的基本要素,包括状态S,奖励函数R,动作空间A;基于最小化有效信息提取的信息处理,定义基于网络的控制决策过程;在无障碍的初始简单场景下,通过学习迭代过程训练机器鼠获得基础行走步态;最后,基于无障碍场景下的训练参数,继续在各种不同障碍场景中训练机器鼠,获得跨越障碍物的适应性步态。本发明针对小型鼠类四足机器人的结构特点设计了状态空间、动作空间、奖励函数,具有更好的控制效果;学习迭代算法采用端对端的控制方式,实现更加简单且在应对复杂地形时表现较好。

首先根据仿生鼠机器人的结构特点构建了基于最小化有效信息提取的信息处理流,定义了强化学习过程中的基本要素,包括状态S,奖励函数R,动作空间A;基于最小化有效信息提取的信息处理,定义基于网络的控制决策过程;

接着在无障碍的初始简单场景下,通过学习迭代过程训练机器鼠获得基础行走步态;最后,基于无障碍场景下的训练参数,继续在各种不同障碍场景中训练机器鼠,获得跨越障碍物的适应性步态。

1. 一种基于强化学习的小型仿生鼠机器人的运动控制方法,其特征在于,包括以下步骤:

首先根据仿生鼠机器人的结构特点构建了基于最小化有效信息提取的信息处理流,定义了强化学习过程中的基本要素,包括状态 S ,奖励函数 R ,动作空间 A ;基于最小化有效信息提取的信息处理,定义基于网络的控制决策过程;

接着在无障碍的初始简单场景下,通过学习迭代过程训练机器鼠获得基础行走步态;最后,基于无障碍场景下的训练参数,继续在各种不同障碍场景中训练机器鼠,获得跨越障碍物的适应性步态;

在具体的控制过程中,对于任意时刻 t ,根据状态空间定义获取当前的状态 s_t ,通过Actor网络 N_a 获取当前采取的动作 a_t 以及选中动作 a_t 的概率 p_t 和奖励值 r_t ;在具体的学习迭代过程中,首先通过控制方法在环境中采样若干 (s_t, a_t, p_t, r_t) 存入经验缓冲池 M ,当 M 已满时,利用经验回放计算损失函数 L ,再通过梯度下降的方法优化Actor网络 N_a 参数 θ_a 和Critic网络 N_c 参数 θ_c 。

2. 根据权利要求1所述的基于强化学习的小型仿生鼠机器人的运动控制方法,其特征在于,基于最小化有效信息提取的信息处理流的流程包括:

S11. 首先基于机器鼠结构,在满足性能前提下选择最精简化的传感数据;速度传感器读数 $V = (V_x, V_y, V_z)$,其中 V_x, V_y, V_z 分别为速度传感器在 x, y, z 轴的读数;姿态传感器读数 Q, Q 是用于表征姿态的四元组;位于机器人各关节处的传感器读数 P_q ,记录了铰链转动角度或滑动器滑动的距离; x_n, y_n, z_n 表示当前位置的三维坐标;

S12. 定义状态空间 $S = \langle V, Q, P_q \rangle$;

S13. 由速度传感器读数通过时间积分计算当前时刻的累计位移 $\vec{d}_n = \vec{d}_{n-1} + V = (x_n, y_n, z_n)$;

S14. 定义跌倒惩罚函数 $f(\vec{d}_n) = \begin{cases} 1 & (z_n < h) \\ 0 & (z_n \geq h) \end{cases}$,常数 h 为高度阈值,用于判断机器人是否跌倒;

S15. 基于累计位移和跌倒惩罚函数,定义奖励函数 $R_n = \alpha(\vec{d}_n - \vec{d}_{n-1}) \cdot \vec{d} - \beta f(\vec{d}_n)$,其中 α, β 为参数表示各部分的重要程度, \vec{d} 为机器人运动目标方向的单位向量;

S16. 通过端对端的方式定义动作空间:

$$A = (q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8)$$

其中,每个数据分量对应机器鼠每个舵机转动角度。

3. 根据权利要求2所述的基于强化学习的小型仿生鼠机器人的运动控制方法,其特征在于,决策控制执行的过程包括:

S21. 初始化Actor网络 N_a ,通过参数 θ_a 定义 N_a ;

S22. 将当前状态 s_t 输入 N_a 得到动作空间的概率分布 $P(s_t; \theta_a)$;

S23. 对概率分布 $P(s_t; \theta_a)$ 采样得到动作 a_t 、选中动作 a_t 的概率 p_t ;

S24. 机器人执行计算出的动作,从环境中得到奖励值 r_t 。

4. 根据权利要求2所述的基于强化学习的小型仿生鼠机器人的运动控制方法,其特征

在于,学习迭代过程具体包括:

S31. 首先在无碍场景初始化Actor网络 N_a 参数 θ_a 和Critic网络 N_c 参数 θ_c ;

S32. 通过决策控制方法获得当前状态 s_t 下的选择的动作 a_t 、选中动作 a_t 的概率 p_t 和奖励值 r_t ;

S33. 将 (s_t, a_t, p_t, r_t) 加入经验缓冲池M;

S34. 重复S32~S33若干次直至缓冲池M已满;

S35. 对经验缓冲池M中的每个状态计算折扣奖励值 $R_t = r_t + \gamma^1 r_{t+1} + \gamma^2 r_{t+2} + \dots + \gamma^n r_{t+n}$, 其中常数 γ 为奖励折扣因子, r_{t+n} 为结束状态或缓冲池M中最后加入的状态;

S36. 在缓冲池M中抽取一个mini-batch用于训练网络 N_a, N_c ;

S37. 重复步骤S36若干次,之后清空经验缓冲池M;

S38. 重复步骤S32~S37若干次,直至训练完成;

S39. 基于训练后得到的参数 θ_a, θ_c ,在不同的障碍场景下重复步骤S32~S38,获得适应不同场景运动的智能体参数。

5. 根据权利要求4所述的基于强化学习的小型仿生鼠机器人的运动控制方法,其特征在于,所述的步骤S36具体包括:

S361. 将mini-batch中的 a_t 输入网络 N_a 得到新的概率分布 $P(s_t; \theta_a)$ 及其交叉熵 $S(s_t; \theta_a)$;

S362. 用新的概率分布 $P(s_t; \theta_a)$ 计算出 a_t 出现的概率 q_t ;

S363. 将 s_t 输入网络 N_c 得到评价价值 V_t ,则优势值函数 $A_t = R_t - V_t$;

S364. 计算损失函数 $L_t^{CLIP+VF+S}(\theta) = \hat{E}_t[L_t^{CLIP}(\theta) - c_1 L_t^{VF}(\theta) + c_2 S(s_t; \theta_a)]$,其中 $\theta = (\theta_a, \theta_c)$, c_1, c_2 是表示重要性的参数, $L_t^{VF}(\theta) = (R_t - V_t)^2$ 为 R_t 与 V_t 的均方误差, $L_t^{CLIP}(\theta) = \hat{E}_t[\min\{\frac{q_t}{p_t} A_t, \text{clip}(\frac{q_t}{p_t}, 1 - \epsilon, 1 + \epsilon) A_t\}]$,常数 $\epsilon \in (0, 1)$,

$$\text{clip}(a, b, c) = \begin{cases} a & (b \leq a \leq c) \\ b & (a < b); \\ c & (c < a) \end{cases}$$

S365. 通过损失函数使用梯度下降法更新 N_c 的参数 $\theta_c < -\theta_c - \xi \frac{\partial(L_t^{CLIP+VF+S})}{\partial(\theta_c)}$ 和 N_a 的参数 $\theta_a < -\theta_a - \xi \frac{\partial(L_t^{CLIP+VF+S})}{\partial(\theta_a)}$,其中 ξ 为学习率。

6. 一种基于强化学习的小型仿生鼠机器人的运动控制系统,其特征在于,包括:

最小化有效信息提取模块:用于根据仿生鼠机器人的结构特点构建了基于最小化有效信息提取的信息处理,定义了强化学习过程中的基本要素,包括状态S,奖励函数R,动作空间A;

决策执行模块:用于基于最小化有效信息提取模块提取的信息处理流,定义基于网络的控制决策过程;

学习迭代模块:用于在无碍的初始简单场景下,通过学习迭代过程训练机器鼠获得基础行走步态;最后,基于无碍场景下的训练参数,继续在各种不同障碍场景中训练机器鼠,获得跨越障碍物的适应性步态;

其中,在具体的控制过程中,对于任意时刻 t ,根据状态空间定义获取当前的状态 s_t ,通过Actor网络 N_a 获取当前采取的动作 a_t 以及选中动作 a_t 的概率 p_t 和奖励值 r_t ;在具体的学习迭代过程中,首先通过控制方法在环境中采样若干 (s_t, a_t, p_t, r_t) 存入经验缓冲池 M ,当 M 已满时,利用经验回放计算损失函数 L ,再通过梯度下降的方法优化Actor网络 N_a 参数 θ_a 和Critic网络 N_c 参数 θ_c 。

7.根据权利要求6所述的基于强化学习的小型仿生鼠机器人的运动控制系统,其特征在于,所述的最小化有效信息提取模块包括:

传感器数据提取模块:用于基于机器鼠结构,在满足性能前提下选择最精简化的传感数据;速度传感器读数 $V=(V_x, V_y, V_z)$,其中 V_x, V_y, V_z 分别为速度传感器在 x, y, z 轴的读数;姿态传感器读数 Q , Q 是用于表征姿态的四元组;位于机器人各关节处的传感器读数 P_q ,记录了铰链转动角度或滑动器滑动的距离; x_n, y_n, z_n 表示当前位置的三维坐标;

状态空间定义模块:用于定义状态空间 $S=\langle V, Q, P_q \rangle$;

累计位移计算模块:用于由速度传感器读数通过时间积分计算当前时刻的累计位移 $\vec{d}_n = \vec{d}_{n-1} + V=(x_n, y_n, z_n)$;

跌倒惩罚函数定义模块:用于定义跌倒惩罚函数 $f(\vec{d}_n) = \begin{cases} 1 & (z_n < h) \\ 0 & (z_n \geq h) \end{cases}$,常数 h 为高度阈值,用于判断机器人是否跌倒;

奖励函数定义模块;用于基于累计位移和跌倒惩罚函数,定义奖励函数 $R_n = \alpha(\vec{d}_n - \vec{d}_{n-1}) \cdot \vec{d} - \beta f(\vec{d}_n)$,其中 α, β 为参数表示各部分的重要程度, \vec{d} 为机器人运动目标方向的单位向量;

动作空间定义模块:用于通过端对端的方式定义动作空间:

$$A = (q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8)$$

其中,每个数据分量对应机器鼠每个舵机转动角度。

8.根据权利要求7所述的基于强化学习的小型仿生鼠机器人的运动控制系统,其特征在于,决策执行模块包括:

初始化模块:用于初始化Actor网络 N_a ,通过参数 θ_a 定义 N_a ;

动作空间概率分布获取模块:用于将当前状态 s_t 输入 N_a 得到动作空间的概率分布 $P(s_t; \theta_a)$;

动作概率选取模块:用于对概率分布 $P(s_t; \theta_a)$ 采样得到动作 a_t 、选中动作 a_t 的概率 p_t ;

奖励值获取模块:用于机器人执行计算出的动作,从环境中得到奖励值 r_t 。

9.根据权利要求7所述的基于强化学习的小型仿生鼠机器人的运动控制系统,其特征在于,学习迭代模块包括:

网络初始化模块:用于在无障碍场景初始化Actor网络 N_a 参数 θ_a 和Critic网络 N_c 参数 θ_c ;

选取模块:用于通过决策控制方法获得当前状态 s_t 下的选择的动作 a_t 、选中动作 a_t 的概率 p_t 和奖励值 r_t ;

经验缓冲池加入模块:用于将 (s_t, a_t, p_t, r_t) 加入经验缓冲池 M ;

第一迭代模块:用于重复运行选取模块和经验缓冲池加入模块,直至缓冲池 M 已满;

折扣奖励值计算模块:用于对经验缓冲池 M 中的每个状态计算折扣奖励值 $R_t = r_t + \gamma$

$\gamma^1 r_{t+1} + \gamma^2 r_{t+2} + \dots + \gamma^n r_{t+n}$, 其中常数 γ 为奖励折扣因子, r_{t+n} 为结束状态或缓冲池M中最后加入的状态;

训练模块: 用于在缓冲池M中抽取一个mini-batch用于训练网络 N_a 、 N_c ;

第二迭代模块: 用于重复运行抽取模块若干次, 之后清空经验缓冲池M;

第三迭代模块: 用于重复运行选取模块、经验缓冲池加入模块、第一迭代模块、折扣奖励值计算模块、抽取模块、第二迭代模块若干次, 直至训练完成;

智能体参数获取模块: 用于基于训练后得到的参数 θ_a 、 θ_c , 在不同的障碍场景下重复运行以上模块, 获得适应不同场景运动的智能体参数。

10. 根据权利要求9所述的基于强化学习的小型仿生鼠机器人的运动控制系统, 其特征在于, 所述的训练模块包括:

输入模块: 用于将mini-batch中的 a_t 输入网络 N_a 得到新的概率分布 $P(s_t; \theta_a)$ 及其交叉熵 $S(s_t; \theta_a)$;

概率计算模块: 用于用新的概率分布 $P(s_t; \theta_a)$ 计算出 a_t 出现的概率 q_t ;

评价价值模块: 用于将 s_t 输入网络 N_c 得到评价价值 V_t , 则优势值函数 $A_t = R_t - V_t$;

损失函数计算模块: 用于计算损失函数 $L_t^{CLIP+VF+S}(\theta) = \hat{E}_t[L_t^{CLIP}(\theta) - c_1 L_t^{VF}(\theta) + c_2 S(s_t; \theta_a)]$, 其中 $\theta = (\theta_a, \theta_c)$, c_1 、 c_2 是表示重要性的参数, \hat{E}_t 表示期望值, θ 表示偏导; $L_t^{VF}(\theta) = (R_t - V_t)^2$ 为 R_t 与 V_t 的均方误差, $L_t^{CLIP}(\theta) =$

$$\hat{E}_t[\min\{\frac{q_t}{p_t} A_t, \text{clip}(\frac{q_t}{p_t}, 1 - \varepsilon, 1 + \varepsilon) A_t\}], \text{常数 } \varepsilon \in (0, 1), \text{clip}(a, b, c) = \begin{cases} a & (b \leq a \leq c) \\ b & (a < b); \\ c & (c < a) \end{cases}$$

参数更新模块: 用于通过损失函数使用梯度下降法更新 N_c 的参数 $\theta_c < -\theta_c - \xi \frac{\partial(L_t^{CLIP+VF+S})}{\partial(\theta_c)}$ 和 N_a 的参数 $\theta_a < -\theta_a - \xi \frac{\partial(L_t^{CLIP+VF+S})}{\partial(\theta_a)}$, 其中 ξ 为学习率。

一种基于强化学习的小型仿生鼠机器人的运动控制方法

技术领域

[0001] 本发明涉及四足机器人运动控制技术领域,更具体地,涉及一种基于强化学习的小型仿生鼠机器人的运动控制方法。

背景技术

[0002] 四足机器人是一种目前应用广泛的机器人,经常被用于许多不同的场景,因此在运动控制过程中实现高效的步态并适应不同的地形是很重要的。目前常用两类方法为基于建模计算的控制方法和基于强化学习的方法。但基于建模计算的方法行为模式单一、适应性较差;已有的强化学习方法大多基于大型刚性机器人,小型柔性机器人,可配置资源远少于大型机器人,无法直接使用传统的强化学习方法。

[0003] 现有专利一种基于DRL的四足机器人路径规划方法,首先通过四足机器人自带的RGB-D相机对环境进行预扫描,基于栅格法对运动环境进行建模得到环境地图,进而分解为一系列具有二值信息的栅格单元,得到初始状态,将其作为深度卷积神经网络的输入,引入注意力机制优化网络结构以解决深度强化学习DQN算法未能很好利用关键局部信息的问题;通过端对端训练,结合四足机器人的运动特点设置奖励函数,利用置信区间上界探索策略对神经网络进行训练,得到四足机器人所能够执行的八个动作的Q值;最后运用人工势场算法不断探测环境中的动态障碍物并进行实时避障,实现有效的四足机器人路径规划。该方法利用视觉传感器的信息处理提出了一种复杂地形下的四足机器人控制方法,在路径规划、障碍跨越等方面取得了很好地效果,但由于小型机器人内部空间较小、可搭载的传感器数目有限,不具备观测环境图像信息的能力,因此该方法难以在小型机器人上使用。

发明内容

[0004] 本发明为克服上述现有技术中的缺陷,提供一种基于强化学习的小型仿生鼠机器人的运动控制方法,具有更好的控制效果,实现更加简单且在应对复杂地形时表现较好。

[0005] 为解决上述技术问题,本发明采用的技术方案是:

[0006] 一种基于强化学习的小型仿生鼠机器人的运动控制方法,包括以下步骤:

[0007] 首先根据仿生鼠机器人的结构特点构建了基于最小化有效信息提取的信息处理流,定义了强化学习过程中的基本要素,包括状态S,奖励函数R,动作空间A;基于最小化有效信息提取的信息处理流,定义基于网络的控制决策过程;

[0008] 接着在无障碍的初始简单场景下,通过学习迭代过程训练机器鼠获得基础行走步态;最后,基于无障碍场景下的训练参数,继续在各种不同障碍场景中训练机器鼠,获得跨越障碍物的适应性步态;

[0009] 在具体的控制过程中,对于任意时刻 t ,根据状态空间定义获取当前的状态 s_t ,通过Actor网络 N_a 获取当前采取的动作 a_t 以及选中动作 a_t 的概率 p_t 和奖励值 r_t ;在具体的学习迭代过程中,首先通过控制方法在环境中采样若干 (s_t, a_t, p_t, r_t) 存入经验缓冲池M,当M已满时,利用经验回放计算损失函数L,再通过梯度下降的方法优化Actor网络 N_a 参数 θ_a 和

Critic网络 N_c 参数 θ_c 。

[0010] 本发明提出了一种基于强化学习的小型仿生鼠四足机器人的运动控制方法,主要克服了现有四足机器人强化学习方法不适合小型机器人的缺点。本控制方法基于小型鼠类机器人的结构特点设计了状态空间、动作空间、奖励函数,使用基于PPO算法设计的学习迭代方法,训练机器鼠自适应地学习各种场景下的运动方法,让小型四足机器人既能自主学习如何行走,又能学会如何通过各种障碍。

[0011] 在其中一个实施例中,基于最小化有效信息提取的信息处理的流程包括:

[0012] S11. 首先基于机器鼠结构,在满足性能前提下选择最精简化的传感数据;速度传感器读数 $V=(V_x, V_y, V_z)$,其中 V_x, V_y, V_z 分别为速度传感器在x、y、z轴的读数;姿态传感器读数 Q, Q 是用于表征姿态的四元组;位于机器人各关节处的传感器读数 P_q ,记录了铰链转动角度或滑动器滑动的距离; x_n, y_n, z_n 表示当前位置的三维坐标;

[0013] S12. 定义状态空间 $S=\langle V, Q, P_q \rangle$;

[0014] S13. 由速度传感器读数通过时间积分计算当前时刻的累计位移: $\vec{d}_n = \vec{d}_{n-1} + V=(x_n, y_n, z_n)$;

[0015] S14. 定义跌倒惩罚函数 $f(\vec{d}_n) = \begin{cases} 1 & (z_n < h) \\ 0 & (z_n \geq h) \end{cases}$,常数 h 为高度阈值,用于判断机器人是否跌倒;

[0016] S15. 基于累计位移和跌倒惩罚函数,定义奖励函数 $R_n = \alpha(\vec{d}_n - \vec{d}_{n-1}) \cdot \vec{d} - \beta f(\vec{d}_n)$,其中 α, β 为参数表示各部分的重要程度, \vec{d} 为机器人运动目标方向的单位向量;

[0017] S16. 通过端对端的方式定义动作空间:

[0018] $A=(q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8)$

[0019] 其中,每个数据分量对应机器鼠每个舵机转动角度。

[0020] 在其中一个实施例中,决策控制执行的过程包括:

[0021] S21. 初始化Actor网络 N_a ,通过参数 θ_a 定义 N_a ;

[0022] S22. 将当前状态 s_t 输入 N_a 得到动作空间的概率分布 $P(s_t; \theta_a)$;

[0023] S23. 对概率分布 $P(s_t; \theta_a)$ 采样得到动作 a_t 、选中动作 a_t 的概率 p_t ;

[0024] S24. 机器人执行计算出的动作,从环境中得到奖励值 r_t 。

[0025] 在其中一个实施例中,学习迭代过程具体包括:

[0026] S31. 首先在无障碍场景初始化Actor网络 N_a 参数 θ_a 和Critic网络 N_c 参数 θ_c ;

[0027] S32. 通过决策控制方法获得当前状态 s_t 下的选择的动作 a_t 、选中动作 a_t 的概率 p_t 和奖励值 r_t ;

[0028] S33. 将 (s_t, a_t, p_t, r_t) 加入经验缓冲池 M ;

[0029] S34. 重复S32 ~ S33若干次直至缓冲池 M 已满;

[0030] S35. 对经验缓冲池 M 中的每个状态计算折扣奖励值 $R_t = r_t + \gamma^1 r_{t+1} +$

[0031] $\gamma^2 r_{t+2} + \dots + \gamma^n r_{t+n}$,其中常数 γ 为奖励折扣因子, r_{t+n} 为结束状态或缓冲池 M 中最后加入的状态;

[0032] S36. 在缓冲池 M 中抽取一个mini-batch用于训练网络 N_a, N_c ;

[0033] S37.重复步骤S36若干次,之后清空经验缓冲池M;

[0034] S38.重复步骤S32~S37若干次,直至训练完成;

[0035] S39.基于训练后得到的参数 θ_a 、 θ_c ,在不同的障碍场景下重复步骤S32~S38,获得适应不同场景运动的智能体参数。

[0036] 在其中一个实施例中,所述的步骤S36具体包括:

[0037] S361.将mini-batch中的 a_t 输入网络 N_a 得到新的概率分布 $P(s_t; \theta_a)$ 及其交叉熵 $S(s_t; \theta_a)$;

[0038] S362.用新的概率分布 $P(s_t; \theta_a)$ 计算出 a_t 出现的概率 q_t ;

[0039] S363.将 s_t 输入网络 N_c 得到评价价值 V_t ,则优势值函数 $A_t = R_t - V_t$;

[0040] S364.计算损失函数 $L_t^{CLIP+VF+S}(\theta) = \hat{E}_t[L_t^{CLIP}(\theta) - c_1 L_t^{VF}(\theta) + c_2 S(s_t; \theta_a)]$,其中 $\theta = (\theta_a, \theta_c)$, c_1 、 c_2 是表示重要性的参数, \hat{E}_t 表示期望值, ∂ 表示偏导; $L_t^{VF}(\theta) = (R_t - V_t)^2$ 为 R_t 与 V_t 的均方误差,

$$L_t^{CLIP}(\theta) = \hat{E}_t[\min\{\frac{q_t}{p_t} A_t, \text{clip}(\frac{q_t}{p_t}, 1 - \varepsilon, 1 + \varepsilon) A_t\}], \text{常数} \varepsilon \in (0,1), \text{clip}(a, b, c) =$$

$$\begin{cases} a & (b \leq a \leq c) \\ b & (a < b); \\ c & (c < a) \end{cases}$$

[0042] S365.通过损失函数使用梯度下降法更新 N_c 的参数 $\theta_c < -\theta_c - \xi \frac{\partial(L_t^{CLIP+VF+S})}{\partial(\theta_c)}$ 和 N_a 的参数 $\theta_a < -\theta_a - \xi \frac{\partial(L_t^{CLIP+VF+S})}{\partial(\theta_a)}$,其中 ξ 为学习率。

[0043] 本发明还提供一种基于强化学习的小型仿生鼠机器人的运动控制系统,包括:

[0044] 最小化有效信息提取模块:用于根据仿生鼠机器人的结构特点构建了基于最小化有效信息提取的信息处理,定义了强化学习过程中的基本要素,包括状态S,奖励函数R,动作空间A;

[0045] 决策执行模块:用于基于最小化有效信息提取模块提取的信息处理流,定义基于网络的控制决策过程;

[0046] 学习迭代模块:用于在无障碍的初始简单场景下,通过学习迭代过程训练机器鼠获得基础行走步态;最后,基于无障碍场景下的训练参数,继续在各种不同障碍场景中训练机器鼠,获得跨越障碍物的适应性步态;

[0047] 其中,在具体的控制过程中,对于任意时刻 t ,根据状态空间定义获取当前的状态 s_t ,通过Actor网络 N_a 获取当前采取的动作 a_t 以及选中动作 a_t 的概率 p_t 和奖励值 r_t ;在具体的学习迭代过程中,首先通过控制方法在环境中采样若干 (s_t, a_t, p_t, r_t) 存入经验缓冲池M,当M已满时,利用经验回放计算损失函数L,再通过梯度下降的方法优化Actor网络 N_a 参数 θ_a 和Critic网络 N_c 参数 θ_c 。

[0048] 在其中一个实施例中,所述的最小化有效信息提取模块包括:

[0049] 传感器数据提取模块:用于基于机器鼠结构,在满足性能前提下选择最精简化的传感数据;速度传感器读数 $V = (V_x, V_y, V_z)$,其中 V_x 、 V_y 、 V_z 分别为速度传感器在x、y、z轴的读数;姿态传感器读数 Q , Q 是用于表征姿态的四元组;位于机器人各关节处的传感器读数 P_q ,

记录了铰链转动角度或滑动器滑动的距离; x_n, y_n, z_n 表示当前位置的三维坐标;

[0050] 状态空间定义模块:用于定义状态空间 $S = \langle V, Q, P_q \rangle$;

[0051] 累计位移计算模块:用于由速度传感器读数通过时间积分计算当前时刻的累计位移 $\vec{d}_n = \vec{d}_{n-1} + V = (x_n, y_n, z_n)$;

[0052] 跌倒惩罚函数定义模块:用于定义跌倒惩罚函数 $f(\vec{d}_n) = \begin{cases} 1 & (z_n < h) \\ 0 & (z_n \geq h) \end{cases}$,常数 h 为高度阈值,用于判断机器人是否跌倒;

[0053] 奖励函数定义模块;用于基于累计位移和跌倒惩罚函数,定义奖励函数 $R_n = \alpha(\vec{d}_n - \vec{d}_{n-1}) \cdot \vec{d} - \beta f(\vec{d}_n)$,其中 α, β 为参数表示各部分的重要程度, \vec{d} 为机器人运动目标方向的单位向量;

[0054] 动作空间定义模块:用于通过端对端的方式定义动作空间:

[0055] $A = (q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8)$

[0056] 其中,每个数据分量对应机器人每个舵机转动角度。

[0057] 在其中一个实施例中,决策执行模块包括:

[0058] 初始化模块:用于初始化Actor网络 N_a ,通过参数 θ_a 定义 N_a ;

[0059] 动作空间概率分布获取模块:用于将当前状态 s_t 输入 N_a 得到动作空间的概率分布 $P(s_t; \theta_a)$;

[0060] 动作概率选取模块:用于对概率分布 $P(s_t; \theta_a)$ 采样得到动作 a_t 、选中动作 a_t 的概率 p_t ;

[0061] 奖励值获取模块:用于机器人执行计算出的动作,从环境中得到奖励值 r_t 。

[0062] 在其中一个实施例中,学习迭代模块包括:

[0063] 网络初始化模块:用于在无阻碍场景初始化Actor网络 N_a 参数 θ_a 和Critic网络 N_c 参数 θ_c ;

[0064] 选取模块:用于通过决策控制方法获得当前状态 s_t 下的选择的动作 a_t 、选中动作 a_t 的概率 p_t 和奖励值 r_t ;

[0065] 经验缓冲池加入模块:用于将 (s_t, a_t, p_t, r_t) 加入经验缓冲池 M ;

[0066] 第一迭代模块:用于重复运行选取模块和经验缓冲池加入模块,直至缓冲池 M 已满;

[0067] 折扣奖励值计算模块:用于对经验缓冲池 M 中的每个状态计算折扣奖励值 $R_t = r_t + \gamma^1 r_{t+1} + \gamma^2 r_{t+2} + \dots + \gamma^n r_{t+n}$,其中常数 γ 为奖励折扣因子, r_{t+n} 为结束状态或缓冲池 M 中最后加入的状态;

[0068] 训练模块:用于在缓冲池 M 中抽取一个mini-batch用于训练网络 N_a, N_c ;

[0069] 第二迭代模块:用于重复运行抽取模块若干次,之后清空经验缓冲池 M ;

[0070] 第三迭代模块:用于重复运行选取模块、经验缓冲池加入模块、第一迭代模块、折扣奖励值计算模块、抽取模块、第二迭代模块若干次,直至训练完成;

[0071] 智能体参数获取模块:用于基于训练后得到的参数 θ_a, θ_c ,在不同的障碍场景下重复运行以上模块,获得适应不同场景运动的智能体参数。

[0072] 在其中一个实施例中,所述的训练模块包括:

[0073] 输入模块:用于将mini-batch中的 a_t 输入网络 N_a 得到新的概率分布 $P(s_t; \theta_a)$ 及其交叉熵 $S(s_t; \theta_a)$;

[0074] 概率计算模块:用于用新的概率分布 $P(s_t; \theta_a)$ 计算出 a_t 出现的概率 q_t ;

[0075] 评价值模块:用于将 s_t 输入网络 N_c 得到评价值 V_t ,则优势值函数 $A_t = R_t - V_t$;

[0076] 损失函数计算模块:用于计算损失函数 $L_t^{CLIP+VF+S}(\theta) = \hat{E}_t[L_t^{CLIP}(\theta) - c_1 L_t^{VF}(\theta) + c_2 S(s_t; \theta_a)]$,其中 $\theta = (\theta_a, \theta_c)$, c_1, c_2 是表示重要性的参数, \hat{E}_t 表示期望值, ∂ 表示偏导; $L_t^{VF}(\theta) = (R_t - V_t)^2$ 为 R_t 与 V_t 的均方误差, $L_t^{CLIP}(\theta) =$

$$\hat{E}_t[\min\{\frac{q_t}{p_t} A_t, \text{clip}(\frac{q_t}{p_t}, 1 - \varepsilon, 1 + \varepsilon) A_t\}], \text{常数} \varepsilon \in (0, 1), \text{clip}(a, b, c) = \begin{cases} a & (b \leq a \leq c) \\ b & (a < b); \\ c & (c < a) \end{cases}$$

[0077] 参数更新模块:用于通过损失函数使用梯度下降法更新 N_c 的参数 $\theta_c < -\theta_c - \xi \frac{\partial(L_t^{CLIP+VF+S})}{\partial(\theta_c)}$ 和 N_a 的参数 $\theta_a < -\theta_a - \xi \frac{\partial(L_t^{CLIP+VF+S})}{\partial(\theta_a)}$,其中 ξ 为学习率。

[0078] 与现有技术相比,有益效果是:本发明提供的一种基于强化学习的小型仿生鼠机器人的运动控制方法,针对小型鼠类四足机器人的结构特点设计了状态空间、动作空间、奖励函数,具有更好的控制效果;本发明的学习迭代算法采用端对端的控制方式,实现更加简单且在应对复杂地形时表现较好。

附图说明

[0079] 图1本发明方法流程示意图。

[0080] 图2是本发明基于最小化有效信息提取的信息处理流程示意图。

[0081] 图3是本发明学习迭代过程流程示意图。

具体实施方式

[0082] 下面将结合本发明实施例中的附图,对本发明实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例仅仅是本发明的一部分实施例,而不是全部的实施例。下面结合具体实施方式对本发明作在其中一个实施例中说明。其中,附图仅用于示例性说明,表示的仅是示意图,而非实物图,不能理解为对本专利的限制;为了更好地说明本发明的实施例,附图某些部件会有省略、放大或缩小,并不代表实际产品的尺寸;对本领域技术人员来说,附图中某些公知结构及其说明可能省略是可以理解的。

[0083] 在本发明的描述中,需要理解的是,若有术语“上”、“下”、“左”、“右”等指示的方位或位置关系为基于附图所示的方位或位置关系,仅是为了便于描述本发明和简化描述,而不是指示或暗示所指的装置或元件必须具有特定的方位、以特定的方位构造和操作,因此附图中描述位置关系的用语仅用于示例性说明,不能理解为对本专利的限制,对于本领域的普通技术人员而言,可以根据具体情况理解上述术语的具体含义。另外,若本发明实施例中有涉及“第一”、“第二”等的描述,则该“第一”、“第二”等的描述仅用于描述目的,而不能理解为指示或暗示其相对重要性或者隐含指明所指示的技术特征的数量。由此,限定有“第一”、“第二”的特征可以明示或者隐含地包括至少一个该特征。另外,全文中出现的“和/或”

的含义为,包括三个并列的方案,以“A和/或B”为例,包括A方案,或B方案,或A和B同时满足的方案。

[0084] 实施例1:

[0085] 本实施例提供一种基于强化学习的小型仿生鼠机器人的运动控制方法,包括以下步骤:

[0086] 首先根据仿生鼠机器人的结构特点构建了基于最小化有效信息提取的信息处理流,定义了强化学习过程中的基本要素,包括状态S,奖励函数R,动作空间A;基于最小化有效信息提取的信息处理,定义基于网络的控制决策过程;

[0087] 接着在无障碍的初始简单场景下,通过学习迭代过程训练机器鼠获得基础行走步态;最后,基于无障碍场景下的训练参数,继续在各种不同障碍场景中训练机器鼠,获得跨越障碍物的适应性步态;

[0088] 在具体的控制过程中,对于任意时刻t,根据状态空间定义获取当前的状态 s_t ,通过Actor网络 N_a 获取当前采取的动作 a_t 以及选中动作 a_t 的概率 p_t 和奖励值 r_t ;在具体的学习迭代过程中,首先通过控制方法在环境中采样若干 (s_t, a_t, p_t, r_t) 存入经验缓冲池M,当M已满时,利用经验回放计算损失函数L,再通过梯度下降的方法优化Actor网络 N_a 参数 θ_a 和Critic网络 N_c 参数 θ_c 。

[0089] 本发明提出了一种基于强化学习的小型仿生鼠四足机器人的运动控制方法,主要克服了现有四足机器人强化学习方法不适合小型机器人的缺点。本控制方法基于小型鼠类机器人的结构特点设计了状态空间、动作空间、奖励函数,使用基于PPO算法设计的学习迭代方法,训练机器鼠自适应地学习各种场景下的运动方法,让小型四足机器人既能自主学习如何行走,又能学会如何通过各种障碍。

[0090] 其中,基于最小化有效信息提取的信息处理的流程包括:

[0091] S11. 首先基于机器鼠结构,在满足性能前提下选择最精简化的传感数据;速度传感器读数 $V = (V_x, V_y, V_z)$,其中 V_x, V_y, V_z 分别为速度传感器在x、y、z轴的读数;姿态传感器读数 Q , Q 是用于表征姿态的四元组;位于机器人各关节处的传感器读数 P_q ,记录了铰链转动角度或滑动器滑动的距离; x_n, y_n, z_n 表示当前位置的三维坐标;

[0092] S12. 定义状态空间 $S = \langle V, Q, P_q \rangle$;

[0093] S13. 由速度传感器读数通过时间积分计算当前时刻的累计位移 $\vec{d}_n = \vec{d}_{n-1} + V = (x_n, y_n, z_n)$;

[0094] S14. 定义跌倒惩罚函数 $f(\vec{d}_n) = \begin{cases} 1 & (z_n < h) \\ 0 & (z_n \geq h) \end{cases}$,常数h为高度阈值,用于判断机器人是否跌倒;

[0095] S15. 基于累计位移和跌倒惩罚函数,定义奖励函数 $R_n = \alpha(\vec{d}_n - \vec{d}_{n-1}) \cdot \vec{d} - \beta f(\vec{d}_n)$,其中 α, β 为参数表示各部分的重要程度, \vec{d} 为机器人运动目标方向的单位向量;

[0096] S16. 通过端对端的方式定义动作空间:

[0097] $A = (q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8)$

[0098] 其中,每个数据分量对应机器鼠每个舵机转动角度。

[0099] 具体的,决策控制执行的过程包括:

[0100] S21.初始化Actor网络 N_a ,通过参数 θ_a 定义 N_a ;

[0101] S22.将当前状态 s_t 输入 N_a 得到动作空间的概率分布 $P(s_t; \theta_a)$;

[0102] S23.对概率分布 $P(s_t; \theta_a)$ 采样得到动作 a_t 、选中动作 a_t 的概率 p_t ;

[0103] S24.机器人执行计算出的动作,从环境中得到奖励值 r_t 。

[0104] 另外,学习迭代过程具体包括:

[0105] S31.首先在无障碍场景初始化Actor网络 N_a 参数 θ_a 和Critic网络 N_c 参数 θ_c ;

[0106] S32.通过决策控制方法获得当前状态 s_t 下的选择的动作 a_t 、选中动作 a_t 的概率 p_t 和奖励值 r_t ;

[0107] S33.将 (s_t, a_t, p_t, r_t) 加入经验缓冲池M;

[0108] S34.重复S32~S33若干次直至缓冲池M已满;

[0109] S35.对经验缓冲池M中的每个状态计算折扣奖励值 $R_t = r_t + \gamma^1 r_{t+1} +$

[0110] $\gamma^2 r_{t+2} + \dots + \gamma^n r_{t+n}$,其中常数 γ 为奖励折扣因子, r_{t+n} 为结束状态或缓冲池M中最后加入的状态;

[0111] S36.在缓冲池M中抽取一个mini-batch用于训练网络 N_a 、 N_c ;

[0112] S37.重复步骤S6若干次,之后清空经验缓冲池M;

[0113] S38.重复步骤S32~S37若干次,直至训练完成;

[0114] S39.基于训练后得到的参数 θ_a 、 θ_c ,在不同的障碍场景下重复步骤S32~S38,获得适应不同场景运动的智能体参数。

[0115] 其中,所述的步骤S36具体包括:

[0116] S361.将mini-batch中的 a_t 输入网络 N_a 得到新的概率分布 $P(s_t; \theta_a)$ 及其交叉熵 $S(s_t; \theta_a)$;

[0117] S362.用新的概率分布 $P(s_t; \theta_a)$ 计算出 a_t 出现的概率 q_t ;

[0118] S363.将 s_t 输入网络 N_c 得到评价值 V_t ,则优势值函数 $A_t = R_t - V_t$;

[0119] S364.计算损失函数 $L_t^{CLIP+VF+S}(\theta) = \hat{E}_t[L_t^{CLIP}(\theta) - c_1 L_t^{VF}(\theta) + c_2 S(s_t; \theta_a)]$,其中 $\theta = (\theta_a, \theta_c)$, c_1 、 c_2 是表示重要性的参数, $L_t^{VF}(\theta) = (R_t - V_t)^2$ 为 R_t 与 V_t 的均方误差, \hat{E}_t 表示期望值, ∂ 表示偏导, $L_t^{CLIP}(\theta) = \hat{E}_t[\min\{\frac{q_t}{p_t} A_t, \text{clip}(\frac{q_t}{p_t}, 1 - \epsilon, 1 + \epsilon) A_t\}]$,常数 $\epsilon \in (0, 1)$,

$$\text{clip}(a, b, c) = \begin{cases} a & (b \leq a \leq c) \\ b & (a < b); \\ c & (c < a) \end{cases}$$

[0120] S365.通过损失函数使用梯度下降法更新 N_c 的参数 $\theta_c < -\theta_c - \xi \frac{\partial(L_t^{CLIP+VF+S})}{\partial(\theta_c)}$ 和 N_a

的参数 $\theta_a < -\theta_a - \xi \frac{\partial(L_t^{CLIP+VF+S})}{\partial(\theta_a)}$,其中 ξ 为学习率。

[0121] 本发明针对小型鼠类四足机器人的结构特点设计了状态空间、动作空间、奖励函数,具有更好的控制效果;本发明的学习迭代算法采用端对端的控制方式,实现更加简单且在应对复杂地形时表现较好;在项目实验中发现强化学习控制方法也有助于提升机器鼠的运动速度,在本发明的机器鼠上学习迭代算法控制的运行速度大约为传统模型控制方法的

4倍。

[0122] 实施例2

[0123] 本实施例提供一种基于强化学习的小型仿生鼠机器人的运动控制系统,包括:

[0124] 最小化有效信息提取模块:用于根据仿生鼠机器人的结构特点构建了基于最小化有效信息提取的信息处理,定义了强化学习过程中的基本要素,包括状态S,奖励函数R,动作空间A;

[0125] 决策执行模块:用于基于最小化有效信息提取模块提取的信息处理流,定义基于网络的控制决策过程;

[0126] 学习迭代模块:用于在无障碍的初始简单场景下,通过学习迭代过程训练机器鼠获得基础行走步态;最后,基于无障碍场景下的训练参数,继续在各种不同障碍场景中训练机器鼠,获得跨越障碍物的适应性步态;

[0127] 其中,在具体的控制过程中,对于任意时刻t,根据状态空间定义获取当前的状态 s_t ,通过Actor网络 N_a 获取当前采取的动作 a_t 以及选中动作 a_t 的概率 p_t 和奖励值 r_t ;在具体的学习迭代过程中,首先通过控制方法在环境中采样若干 (s_t, a_t, p_t, r_t) 存入经验缓冲池M,当M已满时,利用经验回放计算损失函数L,再通过梯度下降的方法优化Actor网络 N_a 参数 θ_a 和Critic网络 N_c 参数 θ_c 。

[0128] 具体的,所述的最小化有效信息提取模块包括:

[0129] 传感器数据提取模块:用于基于机器鼠结构,在满足性能前提下选择最精简化的传感数据;速度传感器读数 $V = (V_x, V_y, V_z)$,其中 V_x, V_y, V_z 分别为速度传感器在x、y、z轴的读数;姿态传感器读数 Q , Q 是用于表征姿态的四元组;位于机器人各关节处的传感器读数 P_q ,记录了铰链转动角度或滑动器滑动的距离;

[0130] 状态空间定义模块:用于定义状态空间 $S = \langle V, Q, P_q \rangle$;

[0131] 累计位移计算模块:用于由速度传感器读数通过时间积分计算当前时刻的累计位移 $\vec{d}_n = \vec{d}_{n-1} + V = (x_n, y_n, z_n)$; x_n, y_n, z_n 表示当前位置的三维坐标;

[0132] 跌倒惩罚函数定义模块:用于定义跌倒惩罚函数 $f(\vec{d}_n) = \begin{cases} 1 & (z_n < h) \\ 0 & (z_n \geq h) \end{cases}$,常数h为高度阈值,用于判断机器人是否跌倒;

[0133] 奖励函数定义模块;用于基于累计位移和跌倒惩罚函数,定义奖励函数 $R_n = \alpha(\vec{d}_n - \vec{d}_{n-1}) \cdot \vec{d} - \beta f(\vec{d}_n)$,其中 α, β 为参数表示各部分的重要程度, \vec{d} 为机器人运动目标方向的单位向量;

[0134] 动作空间定义模块:用于通过端对端的方式定义动作空间:

[0135] $A = (q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8)$

[0136] 其中,每个数据分量对应机器鼠每个舵机转动角度。

[0137] 另外,决策执行模块包括:

[0138] 初始化模块:用于初始化Actor网络 N_a ,通过参数 θ_a 定义 N_a ;

[0139] 动作空间概率分布获取模块:用于将当前状态 s_t 输入 N_a 得到动作空间的概率分布 $P(s_t; \theta_a)$;

[0140] 动作概率选取模块:用于对概率分布 $P(s_t; \theta_a)$ 采样得到动作 a_t 、选中动作 a_t 的概率

p_t ;

[0141] 奖励值获取模块:用于机器人执行计算出的动作,从环境中得到奖励值 r_t 。

[0142] 其中,学习迭代模块包括:

[0143] 网络初始化模块:用于在无碍场景初始化Actor网络 N_a 参数 θ_a 和Critic网络 N_c 参数 θ_c ;

[0144] 选取模块:用于通过决策控制方法获得当前状态 s_t 下的选择的动作 a_t 、选中动作 a_t 的概率 p_t 和奖励值 r_t ;

[0145] 经验缓冲池加入模块:用于将 (s_t, a_t, p_t, r_t) 加入经验缓冲池M;

[0146] 第一迭代模块:用于重复运行选取模块和经验缓冲池加入模块,直至缓冲池M已满;

[0147] 折扣奖励值计算模块:用于对经验缓冲池M中的每个状态计算折扣奖励值 $R_t = r_t + \gamma^1 r_{t+1} + \gamma^2 r_{t+2} + \dots + \gamma^m r_{t+m}$,其中常数 γ 为奖励折扣因子, r_{t+m} 为结束状态或缓冲池M中最后加入的状态;

[0148] 训练模块:用于在缓冲池M中抽取一个mini-batch用于训练网络 N_a 、 N_c ;

[0149] 第二迭代模块:用于重复运行抽取模块若干次,之后清空经验缓冲池M;

[0150] 第三迭代模块:用于重复运行选取模块、经验缓冲池加入模块、第一迭代模块、折扣奖励值计算模块、抽取模块、第二迭代模块若干次,直至训练完成;

[0151] 智能体参数获取模块:用于基于训练后得到的参数 θ_a 、 θ_c ,在不同的障碍场景下重复运行以上模块,获得适应不同场景运动的智能体参数。

[0152] 其中,所述的训练模块包括:

[0153] 输入模块:用于将mini-batch中的 a_t 输入网络 N_a 得到新的概率分布 $P(s_t; \theta_a)$ 及其交叉熵 $S(s_t; \theta_a)$;

[0154] 概率计算模块:用于用新的概率分布 $P(s_t; \theta_a)$ 计算出 a_t 出现的概率 q_t ;

[0155] 评价值模块:用于将 s_t 输入网络 N_c 得到评价值 V_t ,则优势值函数 $A_t = R_t - V_t$;

[0156] 损失函数计算模块:用于计算损失函数 $L_t^{CLIP+VF+S}(\theta) = \hat{E}_t[L_t^{CLIP}(\theta) - c_1 L_t^{VF}(\theta) + c_2 S(s_t; \theta_a)]$,其中 $\theta = (\theta_a, \theta_c)$, c_1 、 c_2 是表示重要性的参数, \hat{E}_t 表示期望值, ∂ 表示偏导, $L_t^{VF}(\theta) = (R_t - V_t)^2$ 为 R_t 与 V_t 的均方误差, $L_t^{CLIP}(\theta) =$

$$\hat{E}_t[\min\{\frac{q_t}{p_t} A_t, \text{clip}(\frac{q_t}{p_t}, 1 - \epsilon, 1 + \epsilon) A_t\}], \text{常数} \epsilon \in (0, 1), \text{clip}(a, b, c) = \begin{cases} a & (b \leq a \leq c) \\ b & (a < b); \\ c & (c < a) \end{cases}$$

[0157] 参数更新模块:用于通过损失函数使用梯度下降法更新 N_c 的参数 $\theta_c < -\theta_c - \xi \frac{\partial(L_t^{CLIP+VF+S})}{\partial(\theta_c)}$ 和 N_a 的参数 $\theta_a < -\theta_a - \xi \frac{\partial(L_t^{CLIP+VF+S})}{\partial(\theta_a)}$,其中 ξ 为学习率。

[0158] 实施例3

[0159] 本实施例提供一种计算机设备,包括:存储器、处理器及存储在所述存储器上并可在所述处理器上运行的计算机程序,所述处理器执行所述计算机程序,以实现上述实施例1中提供的基于强化学习的小型仿生鼠机器人的运动控制方法。

[0160] 实施例4

[0161] 本实施例提供一种计算机可读存储介质,其上存储有计算机程序,该程序被处理器执行,以用于实现实施例1中提供的基于强化学习的小型仿生鼠机器人的运动控制方法。

[0162] 显然,本发明的上述实施例仅仅是为清楚地说明本发明所作的举例,而并非是对本发明的实施方式的限定。对于所属领域的普通技术人员来说,在上述说明的基础上还可以做出其它不同形式的变化或变动。这里无需也无法对所有的实施方式予以穷举。凡在本发明的精神和原则之内所作的任何修改、等同替换和改进等,均应包含在本发明权利要求的保护范围之内。

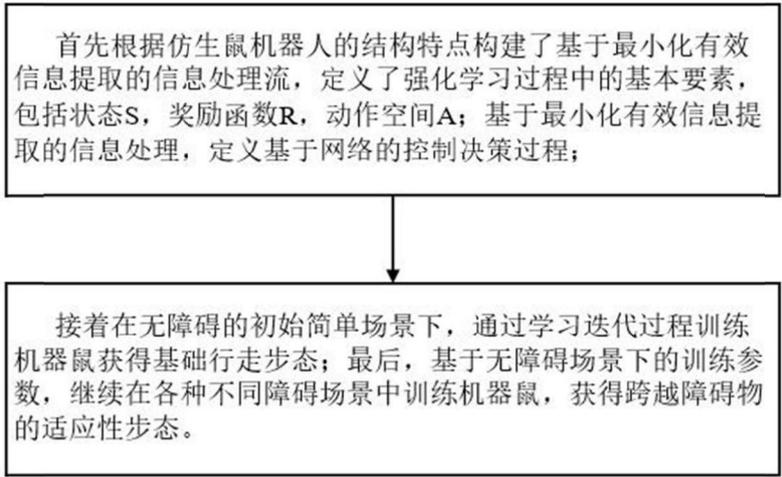


图1

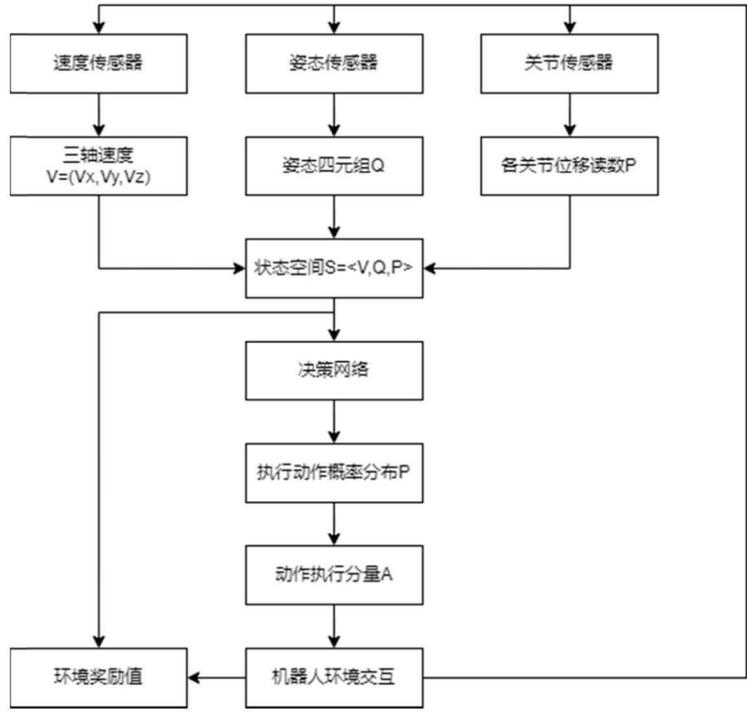


图2

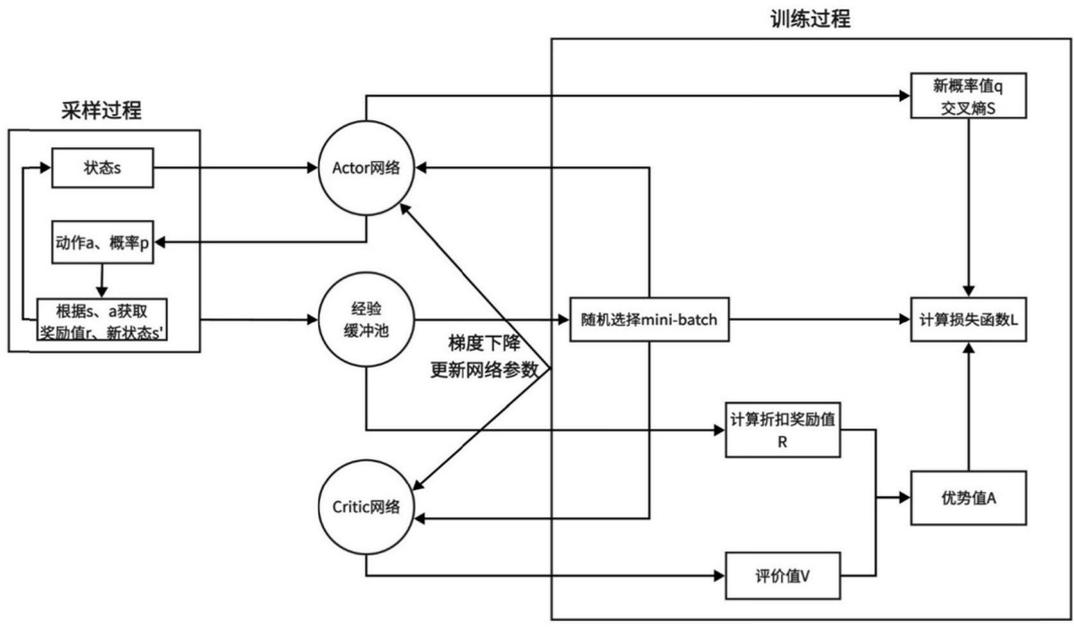


图3