MULE – 多地形与未知负载适应以实现有效的四足运动

Vamshi Kumar Kurva[1], Shishir Kolathaya[2]

***Abstract*— 四足机器人越来越多地被部署于在各种地形上执行负载运输任务。虽然基于模型预测控制（MPC）的方法可以考虑负载变化，但它们通常依赖预定义的步态计划或轨迹生成器，在非结构化环境中限制了其适应性。为了解决这些局限性，我们提出了一种自适应强化学习（RL）框架，使四足机器人能够动态适应不同的负载和多样的地形。该框架由负责基本行走的名义政策和一个学习纠正动作以保持稳定性和改善在负载变化下的命令跟踪的自适应政策组成。我们通过在 Isaac Gym 中的大规模仿真实验以及 Unitree Go1 四足机器人的实际硬件部署验证了所提出的方法。控制器在平坦地面、斜坡和楼梯上进行了测试，包括静态和动态负载变化情况。在所有设置下，我们的自适应控制器始终优于跟踪身体高度和速度命令的控制器，展示了增强的鲁棒性和适应性，而无需明确设计步态或手动调整。**

关键词： *四足动物，腿部运动，强化学习，自适应控制*

I. 介绍

四足机器人的负载能力对于增强其在物流、搜救、军事行动和农业等各个领域的部署至关重要。使这些机器人能够运输大量货物可以显著提高操作效率，减少在危险或难以到达环境中的人工干预需求。尽管在腿部运动方面已经取得了重大进展，特别是在穿越不平坦地形和处理外部干扰方面，但适应未知负载的挑战相对较少被探索。

若干研究试图解决这一问题。[1] 采用了一种在线递归方法，利用接触力和关节角度来估计机器人的惯性参数和基底质心（CoM）。然而，这种方法要求机器人在负载检测过程中停止，限制了其在实时应用中的适用性。[2] 避免直接识别参数，而是学习围绕当前轨迹的局部线性和时变残差模型。该技术支持实时控制，并且在一个 12 公斤的 A1 机器人上携带 10 公斤负载时表现出有效的负载处理能力，尽管对称负载分布的假设最小化了 CoM 的变化。

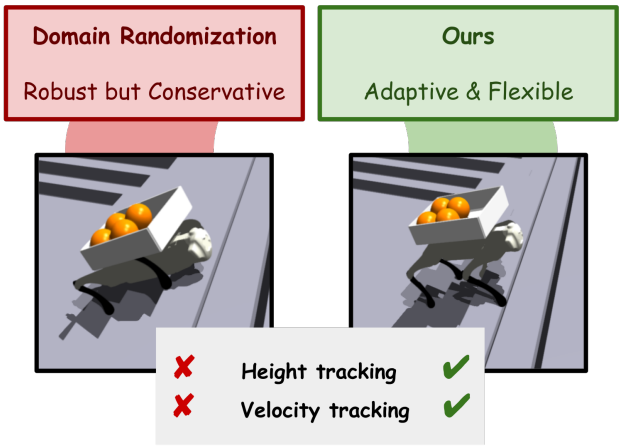

Fig. 1: 域随机化与自适应 RL：尽管域随机化具有鲁棒性但较为保守，我们的自适应方法能够在复杂地形中实现灵活的运动，并改善高度和速度追踪。

其他方法侧重于自适应和鲁棒控制策略。[3] 将 $\mathcal{L}_1$ 自适应控制集成到力控框架中，使 Unitree A1 四足机器人能够稳定地运输 6 公斤的负载。类似地，[4] 引入了一种基于鲁棒优化的鲁棒极小化极大 MPC 策略，以考虑系统不确定性。[5] 在 MPC 框架内结合了控制李雅普诺夫函数（CLF）约束，以确保稳定和自适应的运动，并在 ANYmal 机器人上进行了验证。同时，[6] 将强化学习与模型预测控制相结合，实现了自适应平衡和摆动足反射，使四足机器人能够动态调整负载变化和外部干扰。他们的框架成功展示了在平坦地面上使用 Unitree Go1 机器人处理 7 公斤负载的能力。

上述所有方法都是基于模型的力控制器，根据期望的高度和负载变化来调节站立脚上的地面反作用力（GRFs）。此外，以上所有方法主要在平坦地形或平缓斜坡上显示了负载适应性。这些方法通常将四足动物建模为一个刚体，并计算接触点处应施加的最佳 GRFs，同时低级 PD 控制器跟踪摆动腿的轨迹。为了实现结构化的运动，它们依赖步态或轨迹生成器根据

[1]VK. Kurva is with the Department of Computer Science & Automation, Indian Institute of Science, Bengaluru.

[2]S. Kolathaya is with the Centre for Cyber-Physical Systems and the Department of Computer Science & Automation, Indian Institute of Science, Bengaluru.

This work is supported by ARTPARK and ADB.

Email: stochlab@iisc.ac.in, Project Website: stochlab.com/MULE

步态和速度预定义脚部接触时间表，在摆动腿和站立腿上实施不同的控制策略。基于开关的控制器对站立腿应用力控而对摆动腿使用 PD 控，使系统在非结构化地形上的过早或延迟接触变得敏感，这可能导致不稳定。相比之下，基于 RL 的方法无需依赖预定义的步态时间表 [7] [8] [9] [10] [11] [12] 就能在非结构化地形上实现有效的运动。这些方法直接输出期望的关节位置，并使用 PD 控制器进行跟踪，消除了分阶段切换的需求。通过学习能够隐式适应地形变化和接触条件的策略，基于 RL 的控制器实现了比基于模型的方法更稳健和多样的运动。

基于 RL 方法在非结构化地形行走中的优势，我们提出了一种自适应 RL 框架，该框架使四足机器人能够在各种地形上携带负载。

与依赖于显式模型调整的传统基于模型的方法不同，我们的方法允许四足机器人根据感知到的载荷变化动态调整其行走策略。这消除了对预定义步态时间表的依赖，提供了在处理地形和载荷变化时更大的鲁棒性和灵活性。我们的主要贡献总结如下：

- 我们通过在一个名义策略上增加一个自适应校正策略来引入一种用于在不同负载条件下移动的**自适应 RL 框架**。
- 该框架的训练分为两个阶段：首先在正常条件下训练名义策略，然后是自适应策略，它提供纠正措施而无需显式负载参数估计。
- 我们证明自适应策略在负载场景中显著提高了指令跟踪性能，尤其是在带有额外负载的复杂地形如楼梯上。
- 提出的框架在仿真和硬件上进行了验证，与基线方法相比显示出显著改进。

II. 方法论

A. 预备知识

强化学习（RL）提供了一个框架，通过试错与环境互动来训练自主代理以最大化累积奖励。在四足运动的背景下，RL 将控制问题表述为一个马尔可夫决策过程（MDP），其中机器人学习一种最优策略以实现跨越多样地形的稳定和高效移动。MDP 由元组 $(\mathcal{S}, \mathcal{A}, \mathcal{P}, r, \gamma)$ 定义，其中：$\mathcal{S}$ 表示状态空间，$\mathcal{A}$ 定义了动作空间，$\mathcal{P}$ 表示转换动态，该模型描述了四足动物的状态如何根据应用的动作和环境互动而演变，$r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ 是奖励函数，$\gamma \in (0, 1]$ 是折扣因子，用于平衡即时和长期的奖励。

强化学习的目标是学习一个策略 $\pi_\theta(a_t \mid s_t)$，该策略由 θ 参数化，定义了在给定当前状态 $s_t \in \mathcal{S}$ 时选择动作 $a_t \in \mathcal{A}$ 的概率。智能体通过离散时间步骤 $t = 0, 1, 2, \ldots$ 与环境进行交互，在每个步骤基于当前状态和动作接收一个奖励 $r(s_t, a_t)$。RL 的目标是学习一个最大化期望累计折扣奖励的策略。

$$J(\pi) = \lim_{T \to \infty} \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t r(s_t, a_t)\right]$$

B. 背景

在四足动物 locomotion 中的强化学习的发展取得了显著进展，这标志着创新的方法和框架的出现。来自苏黎世联邦理工学院的基础性工作首次利用了 Nvidia 的 Isaac Gym，这是一种高性能的 GPU 加速模拟器，来训练像 ANYmal [10] 这样的四足机器人。该方法采用了一种师生架构，在这种架构中，教师策略通过特权信息进行训练，但由于依赖于这些信息而无法直接部署在硬件上。相反，开发出一种学生策略来从观察历史推断与特权信息相对应的潜变量。

在此初始框架的基础上，RMA 算法应运而生，专注于在四足机器人中实现实时适应，而不依赖领域知识或参考轨迹 [7]。在此基础上，Walk These Ways [8] 探索了通过操纵步态奖励来实现单一策略下的多种行为，增强了行走的多样性。

DreamWaQ [9] 的引入标志着一个重要的里程碑，因为它采用了不对称的 Actor-Critic 架构，展示了能够开发出稳健且泛化的策略以应对各种地形（如崎岖斜坡和楼梯）的能力。这项工作突显了强化学习在创建能够有效处理复杂环境的适应性运动策略方面的潜力。

域随机化技术通常通过在训练过程中引入机器人参数的小变化来弥合模拟与现实世界部署之间的差距。这有助于开发能够抵御干扰的策略。然而，当参数变化范围过大时，生成的策略往往会过于保守 [13] [14] ，优先考虑鲁棒性而牺牲最优性能。这些限制凸显了需要自适应策略的重要性，这种策略可以动态调整以应对各种条件的变化，例如负载变化，而不是依赖单一通用的方法。

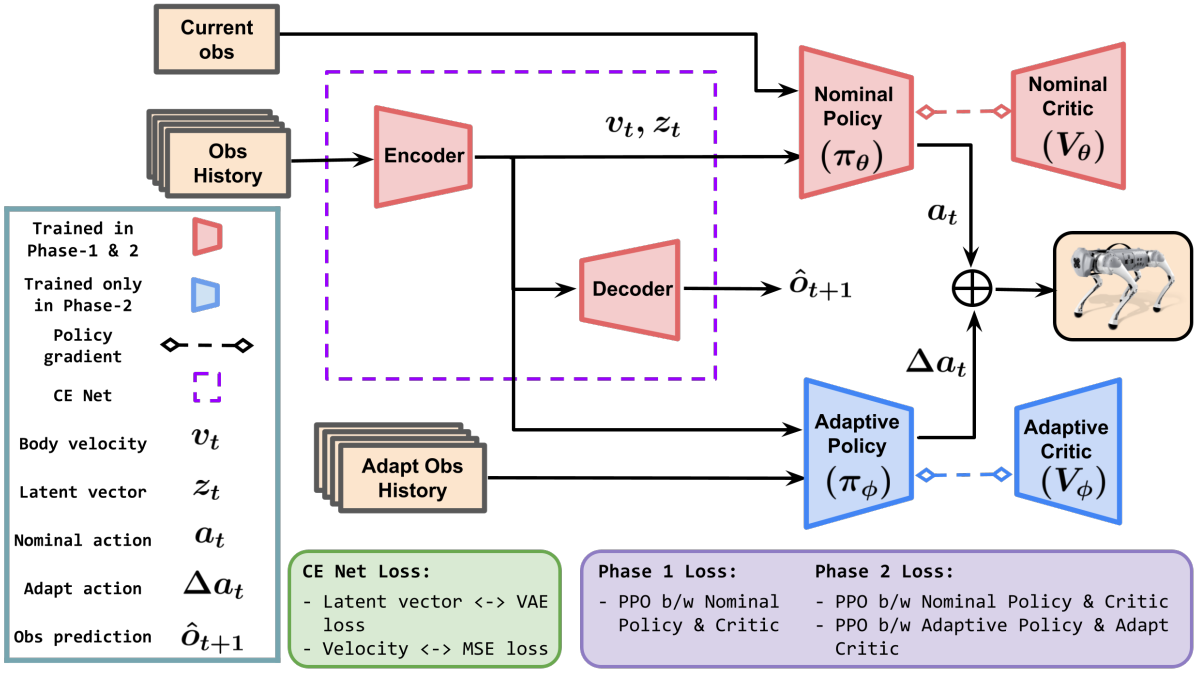

Fig. 2: **提出的框架概述** - 观测历史被编码以获得潜在向量和身体速度使用 CE Net。名义策略和评论家在第一阶段进行训练，而自适应策略和自适应批评者在第二阶段引入以增强对负载变化的适应性。组合动作使在各种负载条件下实现稳健运动成为可能。

C. 自适应 RL 框架

当负载被添加或从机器人上移除时，会导致诸如质量、质心（CoM）和惯性等系统参数发生显著变化，从而改变系统的动力学特性。明确地实时估计这些参数可能会非常具有挑战性并且容易出错。相反，我们提出了一种自适应框架，在该框架中学习一种纠正动作以补偿这些变化。

我们提出的**自适应 RL 框架**（图 2）受到了经典控制中的自适应控制方法和最近的 RL 工作的启发 [15] [16]。它包括两个训练阶段：

- **阶段 1：**我们在正常条件下（没有负载）训练一个名义策略。名义策略负责标准场景中的基本移动和命令跟踪。
- **第二阶段：**我们训练一个自适应策略，在负载条件下提供纠正动作，将增加的负载视为外部干扰。该自适应策略与名义策略协同工作，确保机器人即使在不同负载下也能保持所需的命令跟踪，例如基座高度和速度。

与其显式估计未知的有效载荷参数，自适应策略会动态调整行动以补偿其影响。这种方法使四足机器人能够在其运动策略中进行适应，而无需预定义的步态时间表或显式的模型调整，从而在应对地形和负载变化时提供更大的鲁棒性和灵活性。

D. 第一阶段：名义政策训练

第一阶段的目标是训练一个名义策略以实现在各种地形上的鲁棒行走。

观察结果 观测是一个由以下内容组成的 45 维向量

$$o_t = \begin{bmatrix} \omega_t & g_t & c_t & \theta_t & \dot{\theta}_t & a_{t-1} \end{bmatrix}^T$$

其中 w_t, g_t 是机体角速度和重力矢量，c_t 是机体速度指令，$\theta_t \dot{\theta}_t$ 是关节角度位置和速度，a_{t-1} 是前一动作。

动作 动作是 12 维向量，表示相对于固定站立姿态 θ_{stand} 的期望关节位置，即：

$$\theta_{des} = \theta_{stand} + a_t$$

通过 PD 控制器跟踪期望的关节角度。

奖励 任何给定时间步长的总奖励由

$$r_t(\theta) = \sum_i r_i w_i$$

给出，其中 θ 是名义策略的参数化，i 是奖励组件的索引，而 w_i 是在表中所示的权重。I

编码器 编码器是上下文估计网络（CE 网络）的一部分，它将观测历史 o_t^H 编码为潜在向量 z_t 和身体速度 v_t。使用解码器从编码中重构下一个观测值。β-VAE 用于此自动编码任务。CE 网络通过混合损失函数进行优化，定义如下：

$$\mathcal{L}_{CE} = \mathcal{L}_{est} + \mathcal{L}_{VAE}$$

这些损失直接来自于 [9]。

训练 我们将**名义政策**表示为 π_θ，观测值为 o_t，动作值为 a_t，而**自适应策略** π_ϕ 接受增强的观测值 $\tilde{o}_t$ 并输出修正动作 Δa_t。在第一阶段，我们仅使用近端策略优化（PPO）[17] 训练名义策略 π_θ，而自适应策略 π_ϕ 保持不活跃（即 $\nabla_\phi = 0$）。应用于环境的最终动作是 a_t。名义策略的 PPO 目标是：

$$\mathcal{L}^\theta_{\text{PPO}} = \mathbb{E}\left[\min\left(\rho_t(\theta)\hat{A}^\theta_t,\ \text{clip}\left(\rho_t(\theta), 1-\epsilon, 1+\epsilon\right)\hat{A}^\theta_t\right)\right],$$

其中 $\rho_t(\theta)$ 是当前和旧策略之间的概率比：

$$\rho_t(\theta) = \frac{\pi_\theta(a_t \mid o_t)}{\pi_{\theta_{\text{old}}}(a_t \mid o_t)},$$

而 $\hat{A}^\theta_t$ 是名义策略的优势估计。

E. 第二阶段：在变化负载下的自适应策略训练

在第二阶段，我们同时训练名义策略和自适应策略以应对不同的负载条件。虽然名义策略保留了第一阶段的奖励结构，但自适应策略通过一个优先考虑稳定性和基高度调节及负载适应性的独立奖励形式进行优化。

自适应策略的主要目标是在负载变化的情况下维持机器人所需的基底高度。当由于负载增加导致基底高度低于目标值时，该策略需要在站立脚上施加更大的力以恢复它。估计末端执行器力（即脚部力）对于实现这种纠正行为至关重要。我们使用关节扭矩与所产生的力之间的雅可比关系来估算每个脚的这些力：

$$\tau = J(\theta)^T f \implies f = \left(J(\theta)^T\right)^\dagger \tau,$$

其中 J 是依赖于关节配置 θ 的雅可比矩阵，τ 是施加的关节扭矩向量，f 表示估计的末端执行器力。为了将此信息纳入自适应策略中，我们通过估算脚部力来扩展其观察空间：

$$\text{Adapt obs } \tilde{o}_t = (\text{obs}, f).$$

我们引入了一个 GRF 跟踪奖励，以鼓励自适应策略在基底高度低于目标值时生成更高的 GRFs。

$$r_{\text{GRF}} = 0.75 \times (h > h_{cmd}) + 0.50 \times (h < h_{cmd}) \times \left(\sum_{i=1}^{4} |f_i| > (m_r + m_p)g\right)$$

其中 h 是当前基高，h_{cmd} 是期望基高，f_i 是腿 i 的地面反作用力，而 m_r 和 m_p 分别是机器人和负载的质量。

我们引入动态载荷变化来训练机器人适应不断变化的质量条件。一个轻便的托盘（250 克）安装在机器人的底座上，每个周期开始时，四个球形物体（球）被放置在其中。每个球的初始质量从均匀分布 $[0, 1]$ 千克中抽取样本，导致总载荷可达 4 千克。每 4 秒钟重新从均匀分布 $[0, 2.5]$ 千克中抽取每个球的质量样本。这种连续变化迫使机器人调整其动作以应对不断变化的动力学特性，确保稳定性和准确的命令跟踪。

在此阶段，最终应用于环境的操作是 $a_t + \Delta a_t$。每个策略的 PPO 目标由以下公式给出：

$$\mathcal{L}^\psi_{\text{PPO}} = \mathbb{E}\left[\min\left(\rho_t(\psi)\hat{A}^\psi_t,\ \text{clip}\left(\rho_t(\psi), 1-\epsilon, 1+\epsilon\right)\hat{A}^\psi_t\right)\right],$$

其中，$\rho_t(\psi)$ 是当前和旧策略之间的概率比（$\psi \in \{\theta, \phi\}$），而 $\hat{A}^\psi_t$ 是对应回报估计。

第二阶段的梯度更新由以下给出：

$$\theta \leftarrow \theta - \eta\nabla_\theta \mathcal{L}^\theta_{\text{PPO}},$$
$$\phi \leftarrow \phi - \eta\nabla_\phi \mathcal{L}^\phi_{\text{PPO}}.$$

III. 结果

A. 模拟

我们使用了 Isaac Gym 模拟器来验证我们的控制器。该策略是使用 4096 个代理和 $H = 5$ 的历史大小在 Nvidia RTX A6000 GPU 上进行训练的。所有的演员和评论家网络都包含三个隐藏层，分别有 512、256 和 128 个单元。编码器网络有两个隐藏层，分别有 128 和 64 个单元，而解码器网络也包含两个隐藏层，分别有 64 和 128 个单元。在第一阶段，仅训练名义策略进行了 1000 次迭代。在第二阶段，恢复了来自第一阶段的名义策略权重，并同时对两种策略进行了 500 次迭代的训练。模拟中的关节角度使用预先为 Unitree Go1 训练的动作网络进行跟踪，确保现实的动作行为 [8] [18]。我们对比了命令追踪性能与 DreamWaQ，在基

奖励	Nominal weights (w_i)	Adaptive weights (α_i)
Linear velocity tracking	1.0	0.0
Angular velocity tracking	0.5	0.0
Linear velocity (z)	-2.0	-2.0
Angular velocity (xy)	-0.05	-0.05
Orientation	-0.2	-0.2
Joint accelerations	-2.5×10^{-7}	-2.5×10^{-7}
Joint power	-2.0×10^{-5}	0.0
Body height	-2.0	-2.0
Foot clearance	-0.01	-0.01
Action rate	-0.001	-0.01
Smoothness	-0.01	-0.01
GRF tracking	0.0	2.0

TABLE I: 奖励权重对于名义策略和自适应策略：名义策略优先考虑速度跟踪，而自适应策略侧重于地面反作用力跟踪和身体高度稳定。

础质量随机化范围 $[0, 10]$kg 下，我们将此称为**基线**控制器。

平坦地形 图 4 显示了在不同负载下平坦地形上基准策略和自适应策略的指令跟踪性能。尽管两种策略都以最小误差跟踪前向速度指令，但自适应策略实现了显著更高的高度跟踪精度。基准策略的高度跟踪误差与负载之间存在强烈的正相关性，表明其随着负载增加生成足够地面反作用力的能力有限。

楼梯 图 5 比较了两种策略在楼梯上的性能。基准策略在高负载条件下难以维持速度跟踪，最终停止。高度跟踪误差进一步表明其无法生成足够的地面反应力来保持所需的基础高度。相比之下，自适应策略更有效地跟踪命令，优于基准策略。

图 3 说明了自适应动作和地面反应力如何有助于命令跟踪。随着负载的增加，自适应策略增加了地面反应力的大小和自适应动作范数，使机器人能够保持所需的高度并跟随速度命令。这种正相关表明自适应策略调整地面反应力以补偿较高的负载。有趣的是，尽管自适应策略没有被明确指示何时干预，但它通过训练学会了在名义策略难以跟踪命令时精确地提供纠正动作。这一涌现行为突显了自适应策略根据其特定目标应对不同条件的能力。

B. 硬件部署

实际实验是在 Unitree Go1 机器人上进行的，在其基座上安装了一个 500 克的不锈钢托盘。为了模拟动态载荷变化，将多个 1 千克的铁球放置在托盘内，允许它们自由移动并引起质心的变化。这在整个实验过程中引入了不可预测的载荷动力学。此外，通过在不同试验期间添加和移除 3 千克和 5 千克的圆盘来测试可控的静态载荷变化。这些变化使我们能够评估策略适应逐渐和突然改变载荷条件的能力。由策略指挥的关节角度使用带有 $k_p = 20.0$ and $k_d = 0.5$ 的比例微分控制器进行跟踪。图 6 显示了在逐步增加至 10 千克载荷的情况下，基线控制器与自适应控制器之间的对比。基线控制器难以维持稳定的运动，在较高载荷下表现出明显的脚部打滑和不稳定性，而自适应控制器成功地补偿了额外重量，保持平衡和协调。底部图描绘了随时间变化的自适应动作输出范数，展示了控制器对载荷变化的响应。自适应动作范数中的每次尖峰都对应于新增加的载荷时刻，这说明控制器能够检测并响应质量及动力学的变化。我们还在斜坡上以及楼梯上评估了该控制器，并表明在所有情况下，自适应控制器的表现均优于基线控制器。

IV. 结论

我们提出了一种基于 RL 的自适应控制框架，用于在不同地形和负载配置下实现四足运动。该方法引入了自适应策略，以提供纠正措施来补充名义策略，从而提高整体性能。两个策略都经过训练以优化各自的奖励，并具有某些共同目标，例如稳定性，在它们之间共享。这种共享的奖励结构促进了隐含的合作，即自适应策略互补而不是与名义策略冲突，仅注入必要的最小校正动作以适应意外干扰而不覆盖基础行为。这种模块化设计使系统能够在正常条件下保留名义策略的学习行为，并在检测到偏差时利用自适应策略的快速响应能力。

所提出的自适应 RL 框架成功部署在了 Unitree Go1 机器人上，并在多种地形下进行了评估，包括平坦地面、斜坡和楼梯，在不同静态负载和动态负载场景下，如将自由移动的铁球放置在安装于机器人基座上的托盘中。

在所有测试条件下，我们的策略始终在准确跟踪命令的身体高度和速度方面优于基准控制器，展示了对负载变化和环境变化的优越适应性和鲁棒性。

References

[1] G. Tournois, M. Focchi, A. Del Prete, R. Orsolino, D. G. Caldwell, and C. Semini, “Online payload identification for quadruped

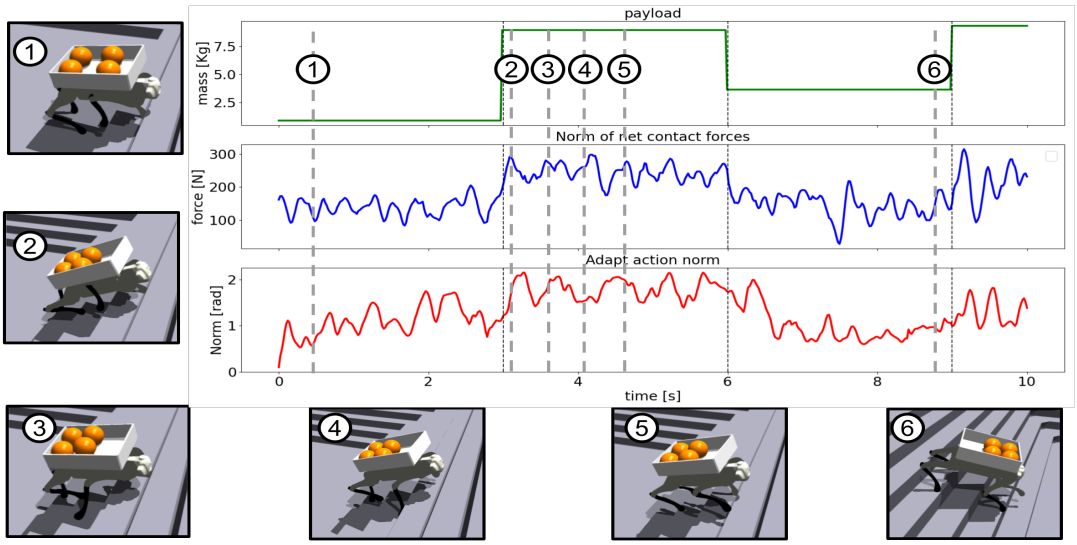

Fig. 3: 四足机器人在楼梯上适应不同负载。（上）具有 6 个阶段的负载质量分布，指示了质量转换。（中）随时间变化的净接触力的范数。（下）自适应动作的范数，展示了控制器对质量和地形转换的响应。快照（1-6）描绘了行进序列中的代表性实例。（2），（3），（4）和（5）显示机器人如何通过生成更高的 GRFs 从重型负载变化中恢复过来。图表还表明适应行动的范数与净接触力之间存在正相关。

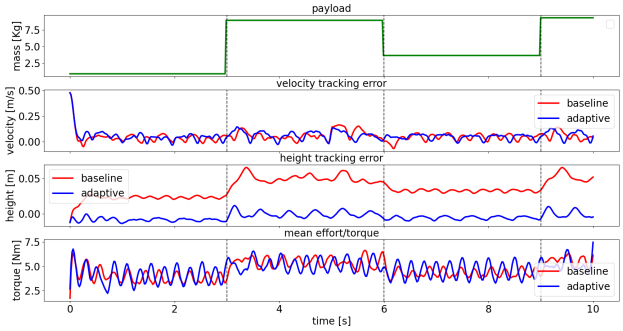

Fig. 4: **平坦地形上基线控制器与自适应控制器的性能比较**顶部部分显示了有效载荷质量分布。两个控制器都表现出类似的速率跟踪和平均扭矩努力，而自适应控制器在高度跟踪方面显著更好。

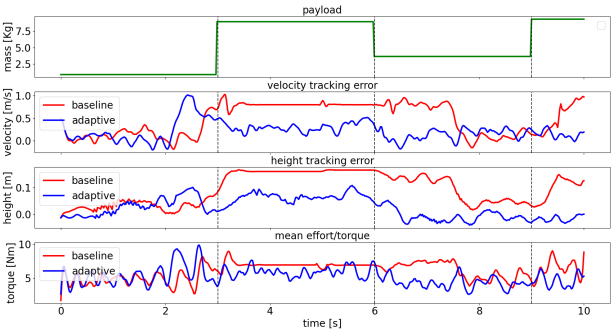

Fig. 5: **楼梯上基准控制器与自适应控制器的性能比较**有效载荷质量分布图显示在顶部。曲线中的平直红色段表示基线控制器未能调整到突然的质量变化，导致机器人完全停止。相比之下，自适应控制器保持稳定的运动状态，在不同负载条件下实现更好的速度和高度跟踪，并显著减少高度跟踪误差和扭矩努力。

robots," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4889–4896, 2017.

[2] Y. Sun, W. L. Ubellacker, W.-L. Ma, X. Zhang, C. Wang, N. V. Csomay-Shanklin, M. Tomizuka, K. Sreenath, and A. D. Ames, "Online learning of unknown dynamics for model-based controllers in legged locomotion," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 8442–8449, 2021.

[3] M. Sombolestan, Y. Chen, and Q. Nguyen, "Adaptive force-based control for legged robots," *CoRR*, vol. abs/2011.06236, 2020.

[4] S. Xu, L. Zhu, H.-T. Zhang, and C. P. Ho, "Robust convex model predictive control for quadruped locomotion under uncertainties," *IEEE Transactions on Robotics*, vol. 39, no. 6, pp. 4837–4854, 2023.

[5] M. V. Minniti, R. Grandia, F. Farshidian, and M. Hutter, "Adaptive clf-mpc with application to quadrupedal robots," *IEEE Robotics and Automation Letters*, vol. 7, p. 565 – 572, Jan. 2022.

[6] Y. Chen and Q. Nguyen, "Learning agile locomotion and adaptive behaviors via rl-augmented mpc," 2024.

[7] A. Kumar, Z. Fu, D. Pathak, and J. Malik, "Rma: Rapid motor adaptation for legged robots," 2021.

[8] G. B. Margolis and P. Agrawal, "Walk these ways: Tuning robot control for generalization with multiplicity of behavior," 2022.

[9] I. M. A. Nahrendra, B. Yu, and H. Myung, "Dreamwaq: Learning robust quadrupedal locomotion with implicit terrain imagination via deep reinforcement learning," 2023.

[10] N. Rudin, D. Hoeller, P. Reist, and M. Hutter, "Learning to walk in minutes using massively parallel deep reinforcement learning," *CoRR*, vol. abs/2109.11978, 2021.

[11] J. Long, Z. Wang, Q. Li, L. Cao, J. Gao, and J. Pang, "Hybrid internal model: Learning agile legged locomotion with simulated robot response," in *The Twelfth International Conference on Learning Representations*, 2024.

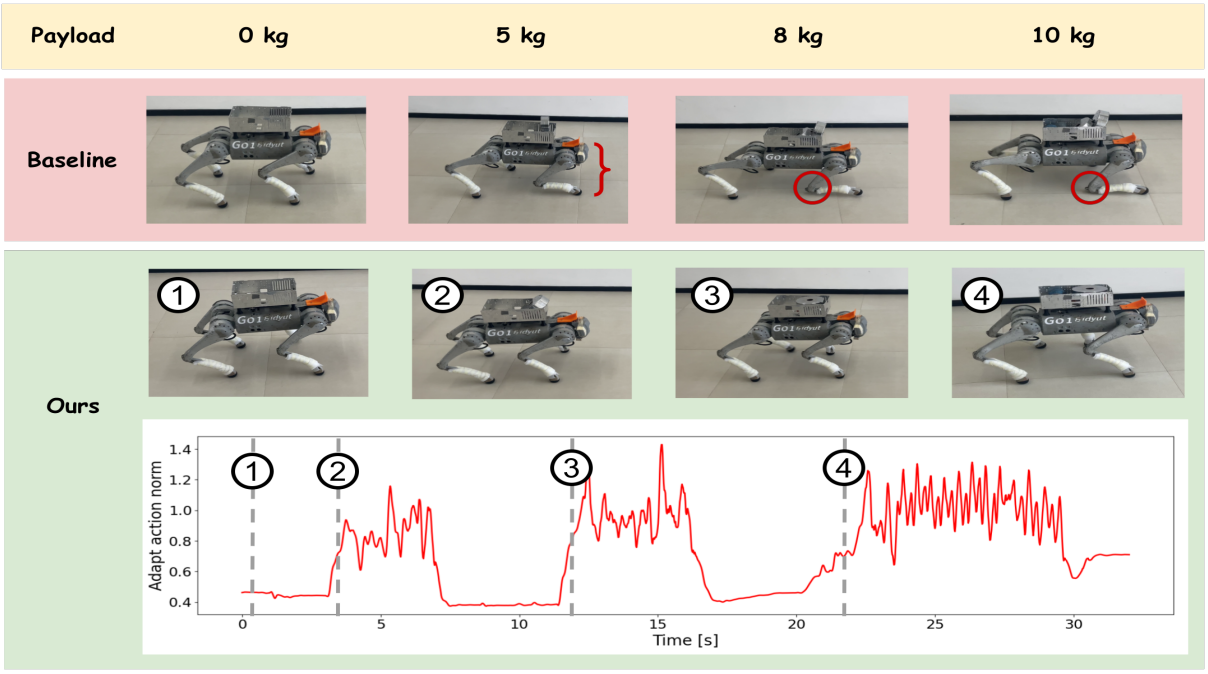

Fig. 6: 在逐步增加负载（0-10 公斤）的情况下，使用基准控制器（顶部行）和提出的自适应控制器（中间行）对比四足动物的运动性能。底部图显示了随时间变化的适应动作范数，捕捉了自适应控制器对负载变化的响应。每个编号标记具体表示添加额外负载使总负载达到所示值的时刻。在两个连续标记之间，负载保持恒定在所示值。适应动作范数接近零的平坦段对应于机器人停止放置或调整负载的短暂停顿。

[12] A. Shirwatkar, N. Saxena, K. Chandra, and S. Kolathaya, "Pip-loco: A proprioceptive infinite horizon planning framework for quadrupedal robot locomotion," 2024.

[13] G. Tiboni, P. Klink, J. Peters, T. Tommasi, C. D'Eramo, and G. Chalvatzaki, "Domain randomization via entropy maximization," 2024.

[14] Y.-H. Lien, P.-C. Hsieh, and Y.-S. Wang, "Revisiting domain randomization via relaxed state-adversarial policy optimization," in *Proceedings of the 40th International Conference on Machine Learning* (A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, eds.), vol. 202 of *Proceedings of Machine Learning Research*, pp. 20939–20949, PMLR, 23–29 Jul 2023.

[15] M. Sung, S. H. Karumanchi, A. Gahlawat, and N. Hovakimyan, "Robust model based reinforcement learning using $\mathcal{L}_1$ adaptive control," 2024.

[16] Z. Li, C. Hu, Y. Wang, Y. Yang, and S. E. Li, "Safe reinforcement learning with dual robustness," 2023.

[17] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017.

[18] J. Hwangbo, J. Lee, A. Dosovitskiy, D. Bellicoso, V. Tsounis, V. Koltun, and M. Hutter, "Learning agile and dynamic motor skills for legged robots," *Science Robotics*, vol. 4, Jan. 2019.