

具身智能中的运动控制强化学习：完整教程

Motion Tracking、Residual RL、Teacher-Student、通用运动控制与 Sim-to-Real

2026-10-19

目录

1. 运动控制强化学习的定位与任务分类	2
2. 运动控制问题的状态、命令与动作	5
3. 方法选择：PPO、SAC、模仿与残差	7
4. Motion Tracking 与动作模仿强化学习	7
5. HOVER、BeyondMimic、SONIC 与通用控制器	9
6. 特权教师、学生蒸馏与动力学自适应	10
7. Residual RL、闭环恢复与安全边界	11
8. 案例：腿式速度控制与全身协调	11
9. 案例：机械臂、灵巧手与接触任务	12
10. 鲁棒控制学习与 Sim-to-Real	14
11. 运动性能、泛化与恢复能力评估	16
12. 论文阅读、工具索引与研究路线	17
参考资料与开源链接	18

课程首页 · English · 仿真工具课程

项目	说明
定位	算法与控制方法（What to Learn）；PPO/SAC 推导与仿真器操作见独立教程
关键问题	如何训练稳定、可泛化、可恢复、能接受多种目标的机器人运动策略？
核验日期	2026-10-19
许可	CC BY-NC-SA 4.0 · Xiong Zeng and contributors

1. 运动控制强化学习的定位与任务分类

阅读说明：示例中的奖励权重、频率和任务设置只用于解释设计原则，并非实机部署的现成安全参数；论文和工具链接可直接点击。

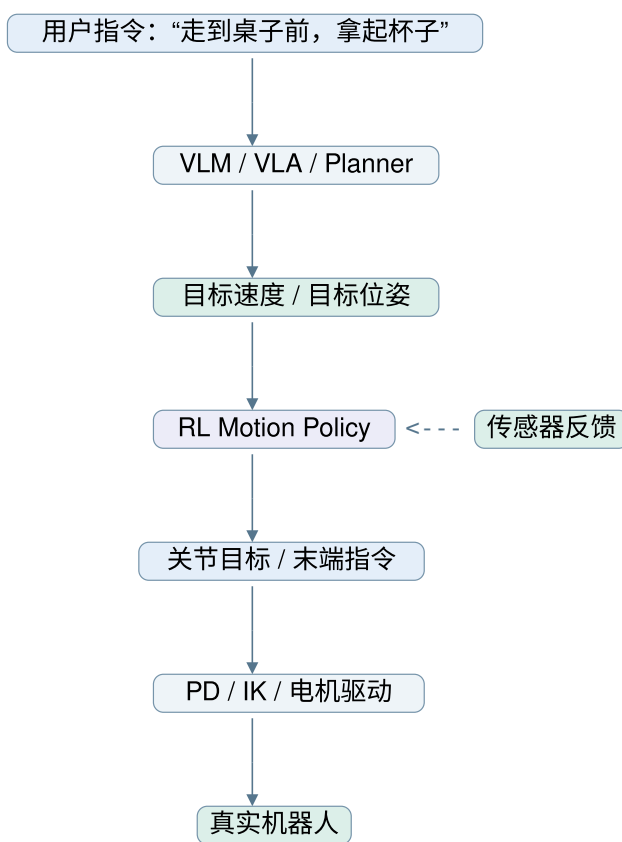
章节知识地图：参考运动/目标命令 → RL 策略 → 安全执行器 → 机器人反馈 → 误差修正。

1.1 运动控制 RL 在具身智能系统中的位置

在具身智能中，可以将机器人控制系统粗略分为三个层次：

层次	主要任务	常见技术
高层：任务规划	决定机器人应该做什么	LLM、VLM、VLA、任务规划
中层：运动规划与技能控制	决定机器人如何运动	Motion Planning、Motion Tracking、RL
底层：执行器控制	让电机准确执行指令	PID、PD、阻抗控制、电流控制

运动控制 RL 主要应用于中层，也可能直接学习部分底层控制。例如，机器人的高层控制器要求“以 0.5 m/s 向前走”，运动策略决定关节动作，底层控制器则负责执行。



一个目标条件运动策略可以写为：

$$a_t = \pi_{\theta}(o_t, c_t),$$

其中， o_t 是关节、IMU、视觉等观测； c_t 是目标速度、目标姿态或技能指令； a_t 是输出的关节位置、力矩或末端动作； π_{θ} 是学习得到的策略。

传统控制通常由工程师显式设计控制律，运动控制 RL 则让策略通过与动力学环境的交互学习控制规律。但它们并不互斥：现代系统经常把 RL 与 PD、IK、MPC、WBC（Whole-Body Control）等技术组合。

1.2 运动控制 RL 与具身基础模型 RL 后训练的区别

维度	运动控制 RL	具身基础模型 RL 后训练
主要目标	学习稳定、灵活的运动技能	提高预训练策略的任务成功率
常见输入	关节、IMU、目标状态等	图像、语言、机器人状态等
常见输出	关节控制指令、末端控制指令	Action Chunk、动作分布等
典型模型	MLP、GRU、Transformer	VLA、Diffusion Policy、Flow Matching
常用方法	PPO、SAC、模仿 RL	PPO、SAC、Offline RL、专门的后训练方法
主要交互来源	大量仿真交互	预训练数据、仿真或真机交互
典型任务	行走、跳跃、爬楼梯、灵巧操作	视觉/语言条件下的操作任务
重点难题	稳定性、鲁棒性、运动性能	泛化、样本效率、在线适应

例如，训练 Unitree G1 在速度指令下行走属于运动控制 RL；对已经能抓取的 $\pi_{0.5}$ 通过 RL 提升新环境抓取成功率属于具身基础模型 RL 后训练。两者边界并不绝对：机械臂操作的低层控制 RL 与 VLA 后训练有时优化的是同一机器人动作，只是模型结构、初始化和目标不同。

1.3 按机器人任务分类

类别	英文	典型任务	主要挑战
腿式运动控制	Legged Locomotion	行走、奔跑、转向	平衡、接触稳定性
高动态运动控制	Agile Locomotion	跳跃、爬楼梯、越障、翻滚	动态接触、强非线性
全身动作跟踪	Whole-Body Motion Tracking	跳舞、动作模仿、轨迹跟踪	全身协调、动作保真度
机械臂操作控制	Manipulation Control	抓取、搬运、插入	接触、精度、力控制
灵巧手操作控制	Dexterous Manipulation	手内旋转、重抓取、工具操作	高维动作、多点接触
移动操作与全身协同	Loco-Manipulation	边走边抓、搬运、推门	行走与操作协同

这些任务并不互斥。人形机器人搬箱子通常同时涉及行走、全身协调和操作。

A. Legged Locomotion：腿式运动控制

目标是根据给定指令稳定运动。常见速度指令为：

$$c_t = (v_x^{\text{cmd}}, v_y^{\text{cmd}}, \omega_z^{\text{cmd}}),$$

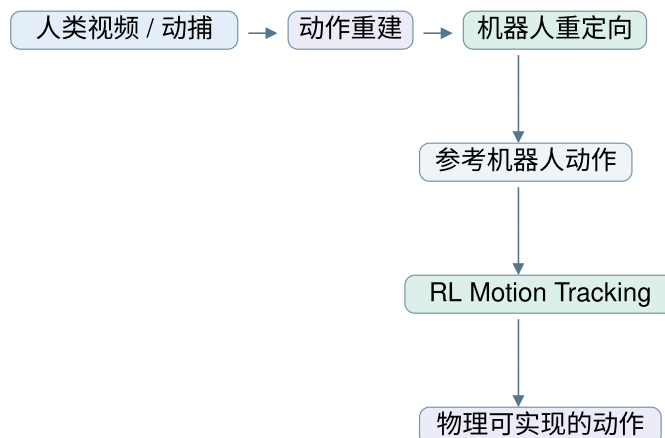
分别对应前后、横向与转向速度。RL 输出关节动作，以跟踪指定速度。四足和双足机器人已经有较成熟的大规模仿真训练范式。代表工作 *Learning to Walk in Minutes Using Massively Parallel Deep Reinforcement Learning* 利用并行仿真训练 ANYmal 运动策略并进行实机验证。[1]

B. Agile Locomotion：高动态运动控制

高动态任务包括跳跃、爬楼梯、跨越障碍、奔跑、跌倒恢复和空中姿态调整。区别于普通行走，策略必须主动利用动力学与接触，例如在爬楼梯过程中协调足部抬升、落脚时序与重心转移。

C. Whole-Body Motion Tracking: 全身动作跟踪

给机器人参考动作 $q^{\text{ref}}(t)$ ，通过 RL 尽量准确地跟踪，同时维持动力学可行性。



代表性路线包括 DeepMimic、AMP、HOVER、BeyondMimic；它们分别涉及参考轨迹模仿、运动先验学习、全身多模式控制和复杂人形动作。[2][3][4]

D. Manipulation Control: 机械臂操作控制

目标是完成抓取、搬运、插入等物体操作。RL 可以输出关节动作，也可以输出末端位置或姿态增量，随后交给 IK、操作空间控制（OSC）或关节控制器。通常更关注末端精度、接触力、成功率、碰撞安全与扰动恢复。

E. Dexterous Manipulation: 灵巧手运动控制

多指灵巧手需要通过多个关节协同控制接触：

$$a_t = [a_t^{\text{finger1}}, a_t^{\text{finger2}}, \dots, a_t^{\text{fingerN}}]^\top.$$

任务包括稳定抓取、手内旋转、重抓取、工具使用以及物体打滑后的恢复。核心困难是高维动作空间与复杂的手指-物体接触动力学。

F. Loco-Manipulation: 移动操作

这类任务将运动与操作联动，例如边走边搬运、扶墙跨越障碍、行走时双手操作物体和推门。重点从单个技能转向对身体、双臂、腿和接触模式的整体协调。

1.4 按训练方法分类

方法	英文	核心思想	适用场景
从零开始强化学习	RL from Scratch	用任务奖励直接探索	基础平衡、行走
动作模仿强化学习	Imitation RL	参考运动奖励引导 RL	跳舞、动作复现
残差强化学习	Residual RL	在已有控制器上学习修正	轨迹修复、接触补偿
分层强化学习	Hierarchical RL	高层选技能、低层控制	多阶段任务
教师-学生训练	Teacher-Student	先训练特权教师，再蒸馏	视觉控制、Sim-to-Real
自适应控制策略	Adaptive RL	根据历史观测适应变化	摩擦、负载、动力学变化

这些方法可以组合：先通过模仿强化学习训练教师，再将策略蒸馏到真实传感器可观测的学生模型，最后进行部署。

2. 运动控制问题的状态、命令与动作

完整环境通常包括：

$$\text{Environment} = \{\text{Observation}, \text{Action}, \text{Reward}, \text{Reset}, \text{Dynamics}\},$$

目标条件任务还需要 Command 或 Goal。实际表现往往比单纯更换 PPO 网络结构更依赖环境建模。

2.1 Observation：机器人能观察到什么？

观测	含义	典型来源
Joint Position	关节位置	编码器
Joint Velocity	关节速度	编码器/估计器
Base Angular Velocity	机身角速度	IMU
Projected Gravity	重力方向在机身坐标系中的表示	姿态估计
Previous Action	上一步控制动作	控制器
Velocity Command	目标速度	上层控制
Contact State	足部/手指接触状态	接触估计
Height Map	地形高度	深度相机或其他传感器

腿式机器人的典型观测可以写为：

$$o_t = [q_t, \dot{q}_t, \omega_t, g_t, a_{t-1}, c_t].$$

要区分仿真可获得的状态与真机实际可用的传感器观测。机身线速度、接触状态和精确物体位姿等在真机上可能需要估计。

Observation History

运动控制往往部分可观测：当前关节角度不能直接确定之前是否滑动、是否受到扰动或执行器是否延迟。可以输入历史窗口：

$$o_t^{\text{hist}} = [o_{t-k}, \dots, o_{t-1}, o_t].$$

随后使用 MLP、GRU 或 Transformer 编码，帮助策略隐式估计隐藏动力学与接触状态。

2.2 Action：强化学习输出什么？

动作空间	输出	特点
Joint Position	关关节目标位置	容易构建稳定基线
Joint Velocity	关关节目标速度	适合速度控制接口
Joint Torque	关关节目标力矩	表达更直接，训练与安全挑战更高
Cartesian Action	末端位置、姿态或速度	适合机械臂

A. Joint Position Control

最常见的一种形式：

$$q_t^{\text{target}} = q^{\text{default}} + s_a a_t, \quad a_t = \pi_\theta(o_t).$$

由底层 PD 执行：

$$\tau_t = K_p(q_t^{\text{target}} - q_t) - K_d\dot{q}_t.$$

对动作跟踪，也可以设定：

$$q_t^{\text{target}} = q_t^{\text{ref}} + s_a a_t.$$

这对应参考运动上的残差动作。优势是减轻 RL 学习底层电机动态的负担。

B. Joint Torque Control

直接输出 $\tau_t = \pi_\theta(o_t)$ ，控制自由度更高，但必须显式考虑电流、力矩、温度、速度和关节限位约束。

C. Cartesian Control

机械臂可输出 $a_t = [\Delta x, \Delta y, \Delta z, \Delta R]$ ，通过 IK 或 OSC 将末端目标转换为关节指令。



其中 ΔR 可由旋转向量等表示，实际应保证旋转参数化与控制频率一致。Isaac Lab 提供多种关节和笛卡尔动作接口。[6]

选型建议

机器人	建议初始动作空间
四足/双足	Joint Position Target
人形动作跟踪	Joint Position Target / Residual
常规机械臂	Cartesian Delta Pose
多指灵巧手	Finger Joint Position Target
机械臂 + 灵巧手	Arm Cartesian Action + Finger Joint Action

策略输出与驱动器最终执行的接口不必相同，例如策略输出末端位姿增量，底层电机仍执行关节位置控制。

2.3 Reward：机器人应该优化什么？

常见拆分：

$$r_t = r_t^{\text{task}} + r_t^{\text{stability}} - r_t^{\text{regularization}}.$$

以双足速度跟踪为例：

$$r_{\text{velocity}} = \exp\left(-\frac{\|v_t - v_t^{\text{cmd}}\|^2}{\sigma_v^2}\right),$$

$$r_{\text{orientation}} = \exp\left(-\frac{\|e_t^{\text{orientation}}\|^2}{\sigma_R^2}\right),$$

$$r_{\text{smooth}} = -\|a_t - a_{t-1}\|^2.$$

可组合成：

$$r_t = w_v r_{\text{velocity}} + w_R r_{\text{orientation}} - w_a \|a_t - a_{t-1}\|^2 - w_\tau \|\tau_t\|^2.$$

实际任务还可能加入：身体高度、足部滑动、碰撞、关节限位、能耗、接触稳定等项。只奖励移动速度可能导致身体前冲、异常高速摆腿或利用仿真接触误差；但惩罚过强则可能使机器人选择不运动。

原则：奖励既要反映任务是否完成，也要约束运动是否稳定、自然、可部署。

2.4 Reset：从什么状态开始训练？

普通行走可以从站立开始；动作跟踪和多阶段操作不能总从同一状态开始，否则难以学习中途恢复。

Uniform Reference State Initialization (Uniform RSI) 从参考运动不同时间点采样：

$$t_0 \sim U(0, T).$$

Event-biased RSI 在起跳、落地、接触切换、开始抓取或释放等关键事件附近增加采样密度。初始化应尽量满足动力学和接触一致性，避免把不合理的跳跃状态强加给策略。

3. 方法选择：PPO、SAC、模仿与残差

本教程只保留与**控制方法选型**直接相关的结论。PPO、GAE、actor/critic 学习与 SAC、Q-learning 的公式推导分别参见课程中独立的 PPO、SAC 讲义。

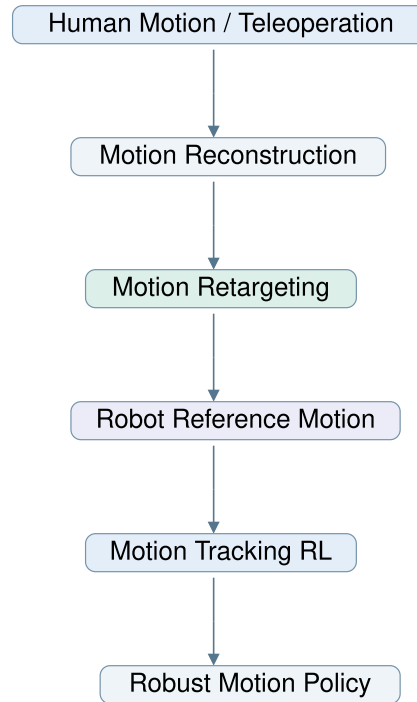
路线	控制问题中的典型位置	主要代价/限制
PPO	仿真并行大、参考动作跟踪、接触动作探索	On-policy，需要重新采样；复用历史数据有限
SAC	连续动作、成本较高的交互、可用 replay 的在线改进	Critic 偏差与外推、稀疏成功信号、并行异步实现更复杂
Imitation + RL	已有动作参考但直接回放动力学不可行	重定向误差、接触错配和参考状态覆盖不足
Residual RL	已有 PD/IK/MPC/模仿策略，需要局部改进或抗扰	基础控制器存在不可逾越的策略上限；必须管理残差范围
Teacher-Student	仿真有特权状态，但真机只能用历史/视觉/本体感知	学生观测别名、蒸馏分布偏移和误差积累

运动控制中的 actor 可写为 $a_t = \pi_\theta(o_{\leq t}, c_t)$ 。对具有安全约束的任务，执行器实际接收的是受限制的 $u_t = \text{Shield}(g(a_t), x_t)$ ，而不是网络输出直接作用于电机。环境/训练器接口归属仿真工具讲义。

4. Motion Tracking 与动作模仿强化学习

复杂跳跃、跳舞、体操动作和高维灵巧操作，直接从零探索可能非常困难，因此可以引入参考运动。

4.1 基本流程



参考数据可以来自动捕、人体视频重建、遥操作、已有控制器以及运动规划。

4.2 Motion Tracking Reward

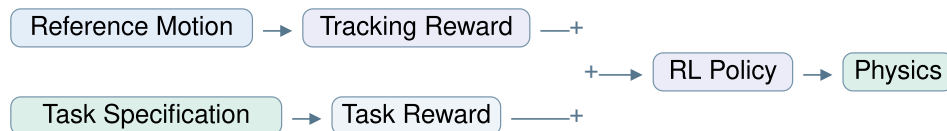
给定参考关节位置 q_t^{ref} ，可以定义：

$$r_{\text{track}} = \exp\left(-\frac{\|q_t - q_t^{\text{ref}}\|^2}{\sigma_q^2}\right).$$

全身任务还可能跟踪 Root Position、Root Orientation、Joint Velocity、End-effector Position、Contact Schedule。仅关节角接近并不保证机器人不会摔倒；需要同时考虑动作相似性、动力学可行性与稳定性。

4.3 DeepMimic

DeepMimic 使用参考运动奖励与任务奖励，帮助物理角色学习复杂动作，是 Motion Tracking RL 的经典工作。[2]



项目主页: <https://xbpeng.github.io/projects/DeepMimic/>

4.4 AMP: Adversarial Motion Priors

传统 Motion Tracking 要求在给定时刻跟踪指定轨迹；AMP 使用判别器从参考动作数据中学习运动风格奖励，使策略在完成任务目标的同时产生符合先验的动作。例如，学习自然的行走风格，而不严格逐帧复制某一条轨迹。[3]

4.5 Motion Tracking 与 AMP 的区别

特性	Motion Tracking RL	AMP
是否需要运动数据	通常需要参考轨迹	需要运动数据
是否显式跟踪某条轨迹	是	不一定
奖励来源	跟踪误差	学到的运动先验奖励
主要目标	准确复现参考运动	产生符合参考分布的运动
典型难点	时间同步、跟踪稳定性	判别器训练、奖励平衡

4.6 从单条轨迹到通用运动控制

HOVER 等工作探索将不同控制命令和动作模式集成到统一的全身控制策略中，使策略不再局限于固定轨迹的逐帧跟踪。[4]

趋势：Single Motion Tracking Policy $\rightarrow$ Multi-Mode / Universal Motion Controller。

4.7 把轨迹跟踪写成可部署的闭环问题

设参考运动含根部位姿、关节角和末端关键点： $m_t^{\text{ref}} = (T_{B,t}^{\text{ref}}, q_t^{\text{ref}}, p_{k,t}^{\text{ref}})$ 。策略并非简单输出 q_t^{ref} ；它需要利用本体感知 o_t 和参考片段 $c_t = (m_{t-h}^{\text{ref}}, \dots, m_{t+H}^{\text{ref}})$ ，在动力学与接触约束下主动补偿误差。

$$r_t = w_p \exp\left(-\frac{\|p_t - p_t^{\text{ref}}\|^2}{\sigma_p^2}\right) + w_q \exp\left(-\frac{\|q_t - q_t^{\text{ref}}\|^2}{\sigma_q^2}\right) + w_R \exp\left(-\frac{d_{SO(3)}(R_t, R_t^{\text{ref}})^2}{\sigma_R^2}\right) - \lambda_u \|u_t\|^2 - \lambda_{\Delta u} \|u_t - u_{t-1}\|^2.$$

这只是奖励结构示例，不是某篇论文的精确实现。必须特别区分世界坐标与根部坐标的跟踪误差；不能用隐含的全局真值来评估一个声称无需全局定位的真机 actor。

RSI 与事件采样：Uniform RSI 增加时序覆盖；接触建立/断开、腾空落地、物体滑移及掉落附近的 event-biased RSI 增加困难状态覆盖。训练时可混合采样，但评估集必须固定且独立，避免训练分布与测试分布混淆。

时序连续性：随机化参考轨迹的初始 SE(3) 位姿时，必须对后续参考位置、方向和目标条件施加一致变换，否则会人为制造不连续命令。

5. HOVER、BeyondMimic、SONIC 与通用控制器

单任务策略与通用控制器的关键区别，不是网络大小，而是**命令接口是否能覆盖多类目标**，**训练分布是否覆盖运动变化**，**闭环策略是否能把目标变成真实运动**。

5.1 从轨迹跟踪到通用技能

一个通用策略写成 $a_t = \pi_\theta(o_{\leq t}, c_t, z_t)$ ，其中 c_t 可是根部速度、头手位姿、关节参考、全身运动片段或高层输出的运动 token， z_t 可是历史压缩的动力学/技能隐变量。关键是使这些条件共享统一且物理一致的动作空间。

代表路线	通用性来源	研究启示
DeepMimic	参考片段 + 物理跟踪奖励 + RSI	先让一条动作动态可行
AMP	从动作数据学习运动分布/风格	运动先验与严格参考跟踪不同
HOVER	多种运动控制模式的统一条件和策略蒸馏	根据 mask 选择根部、末端或关关节目标
BeyondMimic	高质量运动跟踪基元 + 下游 guided diffusion	组合技能与条件生成不等于单纯 PPO 跟踪
SONIC	大规模人类动作数据、运动跟踪预训练、统一 token 接口	对接 VR、人类视频及 VLA 的低层全身控制

5.2 HOVER、BeyondMimic、SONIC 的分工

HOVER: 强调将多种控制目标表达为统一的多模式命令，并通过蒸馏得到可随时切换模式的全身控制策略。难点是命令 mask 的语义一致性与不同模式之间的切换稳定性。

BeyondMimic: 先建立高质量的人形动作跟踪基元，再利用生成式方法进行轨迹组合与任务条件引导。它同时涉及 motion-tracking controller 和较高层的运动生成，不能简单等同为“一个 PPO 运动策略”。

SONIC: 将人类动作跟踪扩展为大规模数据与模型训练，并通过统一的 motion-token 接口连接不同上层条件。这里的 motion token 是低层控制的抽象命令，不意味着模型直接输出未经约束的电机力矩。

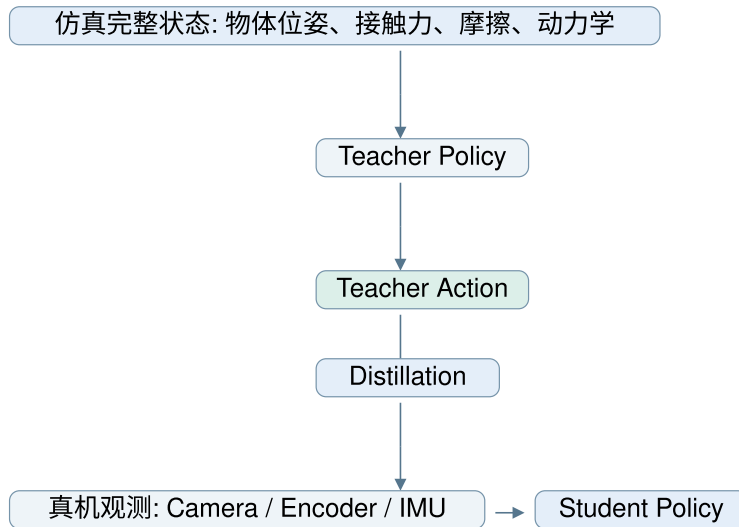
5.3 真正的通用性评估

需要分别测量：① 未见动作的动态跟踪质量；② 不同命令模式切换的延迟与失败率；③ 扰动下恢复能力；④ 不同本体状态与接触模式的覆盖；⑤ 符合时延预算的真机闭环性能。只报告训练轨迹的平均姿态误差无法证明“通用控制器”。

6. 特权教师、学生蒸馏与动力学自适应

6.1 为什么需要教师-学生？

在仿真中可以访问精确物体位置/速度、真实摩擦系数、接触力以及隐藏动力学参数，但这些信息在真机中未必可测。如果 Actor 直接依赖它们，就难以部署。



6.2 标准训练过程

先训练使用特权状态的教师：

$$a_t^T = \pi_T(s_t).$$

再训练只依赖可部署观测的学生：

$$a_t^S = \pi_S(o_t),$$

典型的动作蒸馏损失：

$$\mathcal{L}_{\text{distill}} = \mathbb{E} [\|a_t^S - a_t^T\|^2].$$

学生可借助观测历史和状态估计模块推断隐藏信息。实际蒸馏还需处理教师/学生访问状态分布不同的问题。

6.3 Asymmetric Actor-Critic 与 Teacher-Student 的区别

Asymmetric Actor-Critic: Actor 训练时就只使用可部署观测，而 Critic 可使用 Privileged Information。

Teacher-Student: Teacher Actor 可以使用 Privileged Information，再将教师策略迁移给仅使用有限观测的 Student Actor。所以前者是**训练结构**，后者是**策略迁移方法**。两者可以组合。Isaac Lab 与 RSL-RL 支持相应的观测分组和训练机制。[5][6]

6.4 RMA: Rapid Motor Adaptation

RMA 将动力学适应问题与机器人运动策略结合，利用近期观测历史构建隐式自适应信息，使机器人应对地形、摩擦、负载及未知动力学变化。[7]

论文: <https://arxiv.org/abs/2107.04034>

7. Residual RL、闭环恢复与安全边界

7.1 Residual RL 不是把已有控制器丢掉

给定可解释的名义控制 $u_t^{(0)} = \pi_0(o_t, c_t)$ (PD、IK、MPC 或预训练跟踪策略)，学习有界增量：

$$u_t = \text{clip}\left(u_t^{(0)} + \alpha \Delta u_\theta(o_t, c_t), u_{\min}, u_{\max}\right).$$

残差可以作用在关目标、末端增量、参考轨迹或低层扭矩等不同层。每种选择的频率、可达域与稳定性条件不同；**修正上层参考不等于直接补偿电机力矩**。如要继承名义控制器的稳定性，需对增量施加幅值/变化率约束并验证接触切换下的闭环行为。

7.2 从失败中恢复：要改的不只是奖励

设任务状态在 `approach` → `contact` → `manipulate` → `slip/drop` → `re-localize` → `regrasp` → `continue` 之间转换。若一旦掉落就将 episode 立即终止，策略从训练数据中几乎看不到可恢复状态，也学不到“先重新定位再接近”的行为。

- **状态覆盖：**从扰动前、滑移中、物体落桌和可重抓的多种物理状态开始；只从成功轨迹附近 RSI 并不足够。
- **目标保持：**掉落后重置的应该是局部动作参考或目标估计，而非悄悄改变任务成功条件。
- **观测支持：**需要看见新的物体姿态或从触觉/关节反馈推断抓取失败；纯时间索引跟踪器通常不够。
- **结构设计：**长时程恢复可采用状态机/高层技能切换 + 每技能闭环策略；也可训练单一条件策略，但要明确训练分布和失败成本。
- **指标：**条件恢复率 $P(\text{success} \mid \text{recoverable failure})$ 、重抓次数、恢复时间、安全事件与最终完成率分别报告。

7.3 物理稳定性与 RL 的边界

PD/阻抗/IK/WBC 仍负责执行器快速闭环与约束处理，RL 主要优化其参考、策略和鲁棒修正。对于碰撞、高速或精密装配，必须有独立于策略权重的限位、力矩/速度上限、急停与人工监护。理论稳定性 (ISS/CLF/收缩性等) 不能由训练成功率自动推出。

8. 案例：腿式速度控制与全身协调

8.1 任务定义

让 G1 根据目标速度 $c_t = [v_x^{\text{cmd}}, v_y^{\text{cmd}}, \omega_z^{\text{cmd}}]$ 稳定运动，包括前后、横向移动与转向。

8.2 Observation

建议先使用本体感知: Joint Position、Joint Velocity、Base Angular Velocity、Projected Gravity、Previous Action、Velocity Command。训练楼梯或复杂地形后，再逐步添加 Height Scan、接触估计、观测历史与视觉特征。

8.3 Action

推荐位置目标：

$$q_t^{\text{target}} = q^{\text{default}} + s_a \pi_{\theta}(o_t),$$

底层由 PD 执行，利于复用现有控制接口。

8.4 Reward

奖励项	目标
Linear Velocity Tracking	跟踪前进/横向速度
Angular Velocity Tracking	跟踪转向速度
Orientation Stability	维持合理身体姿态
Action Smoothness	减少动作震荡
Joint Limit Penalty	避免极限关节位置
Foot Slip Penalty	降低足底滑动
Termination Penalty	惩罚摔倒或失败

8.5 Curriculum Learning

阶段	任务	目的
1	站立	平衡
2	低速前进	基本步态
3	前后左右移动	扩大速度范围
4	转向	角速度跟踪
5	外力扰动	稳定与恢复
6	不平地形	接触适应
7	楼梯	复杂地形运动

可以连续扩展命令/地形分布，并不一定每阶段重训单独策略。

8.6 开源实验入口

Unitree RL Lab 基于 Isaac Lab，提供 Unitree 机器人的训练及部署相关代码。[8] 以下命令为该仓库接口的示例，执行前应核对当前版本的环境与任务名称：

第一个研究实验：先复现平地行走，再分别修改 Action Smoothness、Domain Randomization、Observation History 和 Foot Slip Penalty，观察训练收敛、速度误差和扰动恢复是否变化。

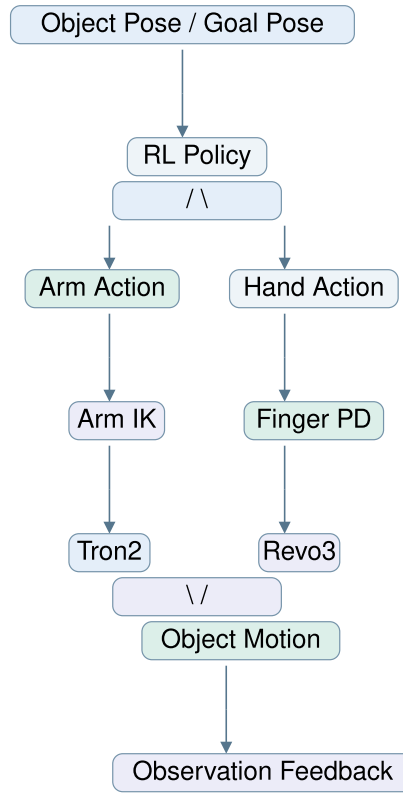
9. 案例：机械臂、灵巧手与接触任务

以 Tron2 机械臂 + Revo3 灵巧手 + 螺丝刀操作 为例，说明如何把腿式运动控制的环境设计方法迁移到复杂接触操作。

9.1 任务定义

1. 从桌面不同位置拾取螺丝刀。
2. 协调机械臂姿态和手指接触。
3. 移动螺丝刀到指定位置。
4. 完成指定角度旋转。
5. 在扰动后保持或恢复抓取。

9.2 建议的控制架构



可以先训练机械臂与手分离的控制器，再逐步扩展到联合策略。

9.3 三种训练路线

路线 A：从零开始训练。直接使用抓取、抬起、旋转和成功奖励：

$$r_t = r_{\text{grasp}} + r_{\text{lift}} + r_{\text{rotate}} + r_{\text{success}}.$$

优点是不依赖参考轨迹，困难在于高维接触探索。

路线 B：Motion Tracking RL。从人类视频或遥操作获得参考： $T_{\text{hand}}^{\text{ref}}(t)$ 、 $q_{\text{finger}}^{\text{ref}}(t)$ 。可以降低探索难度，但单条轨迹不足以覆盖所有物体初始姿态和接触变化。

路线 C：Residual RL。在轨迹跟踪器、IK 或已有策略之上学习受约束的动作修正：

$$a_t = a_t^{\text{base}} + \Delta a_t^{\text{RL}}.$$

例如由基线控制机械臂末端跟踪，RL 修正手指接触，或对臂手都施加有限残差。实际必须约束残差幅值、动作变化率和接触力。

9.4 Observation 设计

$$o_t = [q_{\text{arm}}, q_{\text{hand}}, \dot{q}, \hat{T}_{\text{object}}, T_{\text{goal}}, a_{t-1}].$$

其中 $\hat{T}_{\text{object}}$ 由头部/腕部相机的位姿估计器提供。如果真机无法准确测量物体速度，可用历史观测/状态估计器推断。**Actor** 不应无意中使用只在仿真中有的真实物体状态；Critic 可以采用特权状态。

9.5 Reward 设计

奖励	作用
Reach	靠近目标物体
Grasp	形成稳定抓取
Lift	抬起物体
Position	移动到目标区域
Rotation	完成目标旋转
Stability	减少滑动和掉落
Smoothness	抑制动作震荡
Success	奖励完成完整任务

如果只有最终旋转成功奖励，从随机探索中发现合理抓取可能非常困难。建议使用课程学习与适度分阶段奖励。

9.6 训练路线

阶段	任务	目标
1	已抓住时维持稳定	基本接触控制
2	已抓住时旋转	手内操作
3	接近目标并抓取	抓取能力
4	抓取后旋转	多阶段技能
5	物体初始位姿随机化	泛化
6	加物体位姿估计噪声	视觉鲁棒性
7	加外力扰动	恢复能力
8	真机闭环测试	验证迁移

对于有参考轨迹的任务，可将 Uniform RSI 与 Event-biased RSI 结合，使训练更多覆盖接触建立、释放和物体转动等关键片段。

10. 鲁棒控制学习与 Sim-to-Real

10.1 为什么仿真成功不保证真机成功？

差异类型	典型来源
Mass / Inertia	质量与惯量误差
Friction	地面或物体摩擦不准确
Actuator	电机响应、死区、内部控制
Delay	通信/控制延迟
Observation Noise	编码器、IMU、视觉误差
Contact	接触求解器误差
Geometry	碰撞几何不准确

差异类型	典型来源
Calibration	关节零位、坐标系和外参误差

例如，仿真中电机对目标动作快速响应，而真实硬件有延迟，就可能导致振荡。

10.2 Domain Randomization

训练时随机化环境/动力学参数：

$$\xi \sim p(\xi),$$

其中 ξ 可以包含质量、重心、摩擦、执行器强度、延迟和传感器噪声。

随机化	目的
Mass	适应质量误差
Center of Mass	适应重心偏差
Friction	适应接触表面变化
Motor Strength	适应执行器差异
Action Delay	适应控制延迟
Observation Noise	适应传感器误差
External Push	提高扰动鲁棒性

随机化范围要依据标定数据、硬件规格和物理常识设置，并不是越大越好。

10.3 Sim-to-Sim

可在 Isaac Lab 中训练，再在 MuJoCo 中测试（或反过来），验证策略是否利用了某个仿真器特有的数值行为。



如果同一控制策略在不同仿真器表现差异很大，应优先检查动力学、接触与动作接口。通过 Sim-to-Sim 并不等于实机安全。

10.4 真机部署检查清单

- 观测维度、顺序和标准化参数与训练一致。
- 关节映射、正负方向、单位、坐标系与零位一致。
- Action Scale、PD 增益、控制频率与仿真设定一致或经过验证。
- 位置、速度、力矩、温度设置保护约束。
- 通信中断、推理超时具有安全回退与紧急停止机制。
- 双足机器人配置防跌倒措施；灵巧手注意异常接触力、卡死和物体掉落。
- 初次部署从低风险、低幅值动作开始，逐步扩展任务范围。

进阶：学习真实与仿真之间的动力学差异

ASAP 等工作尝试利用真实机器人数据对齐仿真与真实动力学，改善敏捷全身动作的 Sim-to-Real。[9]

10.5 区分鲁棒学习问题和工程校准问题

误差来源	优先属于哪一层	合理应对
TCP 定义、关节零位、相机内外参	仿真/部署工程	校准与坐标核验；不建议仅靠 DR“学会忽略”
真实执行器延迟、摩擦、质量不确定性	控制策略与动力学适应	系统辨识、DR、历史/潜变量适应、保守残差
目标位姿估计偏差或视觉遮挡	观测与闭环控制	遮挡/噪声训练、时间滤波、主动观察、重定位
出现未训练过的接触/掉落状态	数据覆盖和策略恢复	Event-biased RSI、恢复技能、目标重规划

仿真工具课程给出各类参数怎样在平台里实现与验证；这里重点讨论策略应当怎样利用反馈在未知误差下维持任务能力。不要把 Sim-to-Real 误解为一个能自动修正任意 URDF、外参和通信延迟错误的单独算法。

11. 运动性能、泛化与恢复能力评估

训练 Reward 上升不代表真实任务表现一定改善。必须构建独立测试集和性能指标。

11.1 腿式机器人指标

指标	含义
Success Rate	任务完成率
Fall Rate	摔倒率
Velocity Tracking Error	速度跟踪误差
Orientation Error	姿态误差
Energy Consumption	能耗
Foot Slip	足底滑动
Recovery Rate	扰动恢复率
Generalization	新地形/新参数下的表现

11.2 Motion Tracking 指标

指标	含义
Joint Tracking Error	关节轨迹误差
Root Position Error	根部位置误差
Root Orientation Error	根部方向误差
End-effector Error	末端误差
Contact Consistency	接触一致性
Motion Completion Rate	动作完成率
Recovery Rate	扰动后的恢复率

11.3 灵巧手操作指标

指标	含义
Task Success Rate	任务成功率
Object Pose Error	物体位姿误差
Grasp Stability	抓取稳定性
Object Drop Rate	物体掉落率
Contact Slip	滑动情况
Recovery Rate	扰动恢复率
Completion Time	任务完成时间
Generalization	新物体位姿下的表现

公平对比原则：相同环境与任务、相同交互预算、多个随机种子、未见过的测试状态，以及与训练奖励独立的性能指标。尤其要区分训练表现、仿真泛化和真机表现。

12. 论文阅读、工具索引与研究路线

这里只给出选用开源框架的**研究侧理由**，安装、环境配置、批量后端、性能调试和版本兼容集中在《Isaac Sim、Isaac Lab 与 MuJoCo》教程。

控制问题	建议阅读的算法材料	框架入口
PPO 速度控制/跟踪	On-policy rollout, GAE, 奖励/约束	RSL-RL + Isaac Lab
全身参考运动与蒸馏	Reference retargeting, RSI, DAgger	HOVER, SONIC
复杂接触与灵巧操作	动作空间、接触奖励、滑移/重抓	MuJoCo Playground / Isaac Lab
基于人类动作的通用运动控制	大规模 tracking、生成/组合、统一命令	BeyondMimic, SONIC

边界：训练平台并不规定必须使用 PPO，也不保证算法支持所有后端。阅读开源实现时应把策略输入/输出、教师权限、损失函数和真机闭环频率与物理平台分开分析。

论文	年份	核心主题	重点学习
DeepMimic	2018	动作模仿 RL	参考运动奖励, RSI
AMP	2021	对抗运动先验	从运动数据学习风格
RMA	2021	快速自适应	历史观测与动力学适应
Learning to Walk in Minutes	2022	并行 PPO	大规模运动训练
Learning Robust Perceptive Locomotion	2022	感知运动控制	地形感知与迁移
HOVER	2025	全身多模式控制	通用控制命令
ASAP	2025	动力学差异对齐	Sim-to-Real
BeyondMimic	2025	人形复杂动作	动作跟踪与技能泛化
MuJoCo Playground	2025	GPU 机器人学习	现代实验平台

建议优先精读四项：

1. **Learning to Walk in Minutes**：并行 PPO、奖励、课程学习与 Sim-to-Real。[1]
2. **DeepMimic**：参考动作、动作模仿奖励、RSI、物理动作跟踪。[2]
3. **RMA**：特权状态、历史观测、自适应以及部署。[7]
4. **HOVER**：全身控制、多命令模式、动作跟踪与蒸馏。[4]

12.1 阶段一：基础运动控制 RL

目标：训练一个能够稳定行走的双足/四足机器人。

学习 Isaac Lab、RSL-RL、PPO、Observation/Action/Reward、Domain Randomization。先复现 G1 等平台的速度跟踪，再做奖励和观测消融。

12.2 阶段二：Motion Tracking RL

目标：跟踪参考动作，同时具备物理可行性与恢复能力。

学习 Motion Retargeting、Tracking Reward、RSI、Motion Curriculum、Residual RL。从单条参考动作开始，再逐步增加初始状态扰动和外部扰动。

12.3 阶段三：Teacher-Student 与 Sim-to-Real

目标：策略不依赖仿真中特有的完整状态。

学习 Asymmetric Actor-Critic、Privileged Information、蒸馏、Observation History 与 Domain Randomization。训练特权教师，将其迁移给可部署学生，并开展跨仿真器测试。

12.4 阶段四：复杂运动控制

路线	典型任务
Humanoid Locomotion	爬楼梯、越障、跳跃、跌倒恢复
Dexterous Manipulation	手内旋转、工具操作、扰动恢复

深入研究接触动力学、任务课程、强鲁棒性和真机闭环策略。

12.5 阶段五：通用运动控制器

探索一个支持多种命令的条件策略：

$$a_t = \pi_{\theta}(o_t, c_t).$$

其中 c_t 可编码目标速度、目标位姿、参考运动片段、目标物体运动或技能条件。对应方向包括 Goal-conditioned Motion Control、Universal Motion Tracking、Hierarchical Motion Policies、Residual Motion Adaptation 和 Learning-based Whole-Body Control。

参考资料与开源链接

以下资料主要为教程中提及的论文主页、论文页面和官方开源仓库。论文年份与任务名称应以其最终发表版本/仓库当前版本为准。

1. Rudin et al., *Learning to Walk in Minutes Using Massively Parallel Deep Reinforcement Learning* (2022)
2. Peng et al., *DeepMimic* (2018)
3. Peng et al., *AMP: Adversarial Motion Priors* (2021)
4. HOVER: Versatile Neural Whole-Body Controller for Humanoid Robots; 论文
5. RSL-RL 官方仓库
6. Isaac Lab 官方仓库; 环境目录
7. RMA: Rapid Motor Adaptation for Legged Robots
8. Unitree RL Lab 官方仓库
9. ASAP: Aligning Simulation and Real-World Physics
10. MuJoCo Playground 官方仓库
11. legged_gym 官方仓库

12. skrl 官方仓库

文档版本：2026-10-15。示例参数用于教学，不能直接视为真实机器人安全部署配置。

13. BeyondMimic: From Motion Tracking to Versatile Humanoid Control via Guided Diffusion

14. SONIC: Supersizing Motion Tracking for Natural Humanoid Whole-Body Control

15. SONIC 官方研究页面

16. HOVER 论文及多模式控制介绍

课程边界：Isaac Sim、Isaac Lab、MuJoCo 的建模、安装、执行器对齐、并行训练和性能排错统一见配套工具教程。PPO/GAE 与 SAC/Q-learning 的详细推导见本课程体系独立讲义。