arXiv:2303.04705v1 [cs.RO] 8 Mar 2023

使用模块化强化学习架构进行灵巧触觉手内操作

Johannes Pitz、Lennart R¨ostel、Leon Sievers 和 Berthold B¨auml

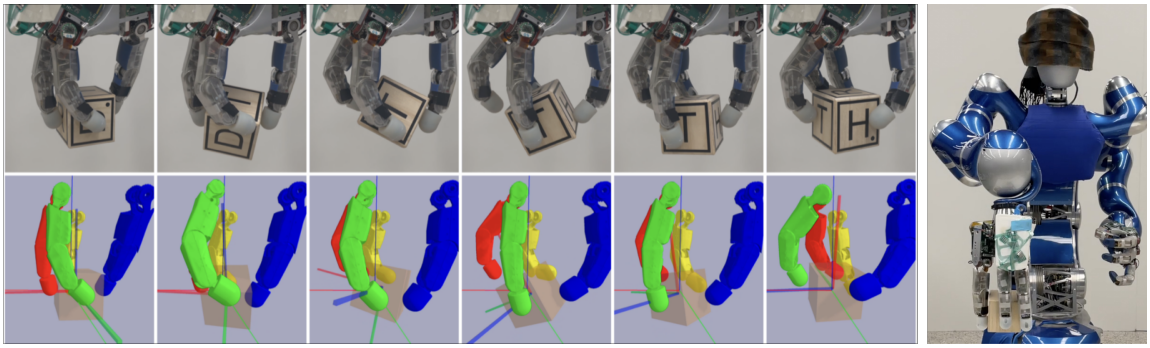

图 1：力矩控制的 DLR-Hand II [1] 正在执行手内操作。立方体被旋转至目标姿态，该姿态可通过三次$\pi/2$ 旋转到达（所有 24 种目标姿态见图 6；此处从 3 开始，到达 20）。任务完全通过触觉完成，不使用外部传感器（没有摄像机——Agile Justin [2] 被蒙住了眼睛）。下排显示的是由训练好的深度可微粒子滤波器估计出的立方体状态。

摘要—使用多指机器人手进行灵巧手内操作是一项具有挑战性的任务，尤其是在手掌朝下、需要始终保持力闭合且不使用外部传感器的情况下。对于将物体重新定向至给定目标姿态的任务（而非使其无限绕某一轴旋转），缺少外部传感器又带来了一项根本性挑战：必须持续估计物体状态，例如判断何时达到目标。本文展示了使用力矩控制的 DLR-Hand II，将立方体重新定向至栅格中的任意一种 $\pi/2$24 种可能目标姿态是可行的。该任务通过模块化深度强化学习架构在仿真中学习：实际策略仅使用 0.5 秒的短观测时间窗口，但会将立方体状态作为显式输入；该状态由深度可微粒子滤波器估计，而滤波器则通过运行策略生成的数据进行训练。在仿真中，即使采用显著的域随机化，我们仍达到了 92% 的成功率。通过在真实机器人系统上进行零样本仿真到现实迁移，全部 24 种目标姿态均能以较高成功率实现。（网页：dlr-alr.github.io/dlr-tactile-manipulation）

I. 引言

许多重要的应用领域，例如工业制造或家务，至今仍需要由人类完成。其中一个重要原因是这些工作需要灵巧的精细操作。具体而言，手内操作，即在手中重新调整物体的方向，是一项经常需要执行但颇具挑战的任务，因为它涉及复杂的多接触动力学、多指手众多自由度（DOF）的协调，以及对被操作物体状态的估计。在本文中，我们扩展了此前关于学习纯触觉手内操作的研究 [3]

作者来自德国航空航天中心机器人与机电一体化研究所、慕尼黑工业大学和代根多夫工业大学。联系方式： {johannes.pitz |berthold.baeuml}@dlr.de

将力矩控制手从只能绕单一轴旋转立方体，扩展到能够将立方体重新调整为 $\pi/2$栅格中的24种可能方向中的任意一种（见图1）。我们在真实场景中完成这项任务：将手倒置，以确保始终保持力闭合；不使用外部传感器（如摄像机），而仅使用手指集成的位置和力矩传感器。我们从零开始，采用模块化深度强化学习架构来学习该任务，其中，操作策略的学习与立方体状态估计器的学习相互分离。图2和图3概述了机器人与任务的设置，以及控制架构。

A. 相关工作

在一项开创性工作中，OpenAI [4] 首次展示了在手掌朝上的手上对立方体进行灵巧手内操作。为了执行目标导向策略，研究人员根据多个摄像机拍摄的图像估计所需的立方体状态。近期，Handa 等人 [5] 展示了使用更简单摄像机配置完成同一任务的真实世界结果。与此不同，在我们研究的设置中，由于手朝下，必须始终保持力闭合。此外，我们仅使用手部集成的位置和力矩传感器，不借助视觉反馈；这种设置更贴近现实，但也使状态估计问题显著更加困难。

在我们之前的工作中 [3], , 我们采用了与本文相同的设置，但任务是让立方体绕垂直轴连续旋转。Qi 等人 [6] 甚至展示了如何对多种物体进行此类操作。对于本文介绍的任务，物体不仅需要绕任意轴旋转，还需要使用状态估计器跟踪立方体状态，因为必须达到特定的目标姿态。Khandate 等人 [7] 则完全在仿真中展示了触觉

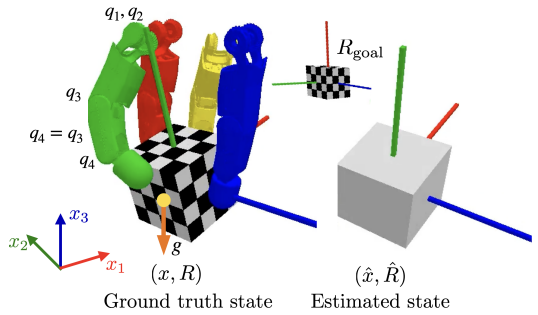

图 2：机器人和任务设置。四指手的每根手指有三个主动 q_1，q_2，q_3 和一个被动关节 q_4。立方体应从初始姿态旋转到目标姿态 R_{goal}。我们将当前姿态 R 与目标姿态 R_{goal} 之间的（正）夹角定义为$\theta = d(R_{\text{goal}}, R)$。在训练步骤 S1-S3（参见表 III）中，向策略提供立方体状态的真实值；而在测试期间，则使用深度粒子滤波器估计状态。

手内操作：在保持力闭合的同时，简单的几何形状可以绕 x 轴、y 轴和 z 轴连续旋转。此外，Chen 等人 [8] 仅在仿真中，并利用视觉输入进行状态估计，展示了在部分实验中手倒置时对多种物体进行基于目标的重新定向。关于仿真中大量手内操作研究的更全面讨论，请参阅我们之前的论文 [3]。

B. 贡献

- 我们证明，仅使用触觉信息（仅使用力矩和位置传感器）进行手内操作，即可将立方体重新定向至 24 种目标姿态中的任意一种。
- 我们设计了一种模块化学习架构，分别训练用于实际手指控制的策略（将立方体状态作为观测的一部分），以及一种基于可微粒子滤波器的估计器 [9]，该估计器仅使用力矩和位置传感器读数。
- 我们展示了如何通过迭代优化策略和估计器、修改奖励，并利用域随机化和非对称观测，成功结合这两个模块。
- 在仿真中，尽管进行了显著的域随机化，成功率仍达到 92%。通过零样本仿真到现实迁移，真实的 DLR-HandII 成功实现了所有目标姿态。

II. 模块化学习架构

A. 动机

尽管机器学习的趋势是通过一个庞大的神经网络模型进行端到端学习，但有时将架构模块化仍然很有用，例如，这有助于在执行过程中，尤其是在学习过程中，深入了解系统。为了解决上述手内操作任务，控制器需要在执行过程中持续跟踪物体状态。否则，就无法区分立方体的多种不同 $\pi/2$ 方向。现在，虽然可以训练一个循环网络控制器在仿真中解决这一任务，但如果策略无法在真实机器人上运行，就几乎不可能分析问题出在哪里。

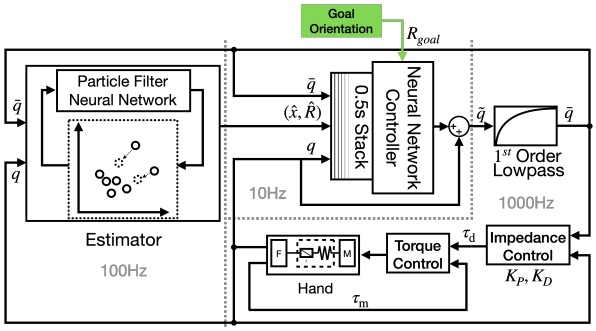

图3：系统总体架构。DLR-Hand II [1] 利用集成力矩传感器提供关节级高保真阻抗控制，充当可配置的弹簧阻尼系统。通过将期望关节角度作为统一接口，以策略网络适中的更新频率（10 Hz），既能实现自由空间运动，也能精细控制接触力。借助低通滤波器，系统的动力学被显式降低，以便于仿真到现实迁移。策略网络的输入根据期望目标姿态、测得的关节角度以及估计的立方体位姿计算得出，并堆叠过去 0.5 秒的数据。立方体状态（位置和姿态）由一个通过学习得到的可微粒子滤波器估计，该滤波器利用测得的和指令发送的手指角度，从而获得接触信息。

B. 模块

整体架构如图3所示，由以下组件组成（详情将在相应章节中介绍）。

1) 状态估计器：在考虑执行目标导向的重新定向所需的信息时，立方体状态显然是抽象层的一个合适选择，并已被用于手内操作 [4, 8, 10]。但与这些工作不同，我们希望以纯粹的触觉方式估计物体状态。为此，我们采用了一种可微粒子滤波器的变体 [11, 12]，这是一种基于学习的方法，我们已证明它可用于手内操作领域，见 R¨ostel 等人 [9]。

2) 网络控制器：通过使用独立模块估计立方体状态，我们可以为控制器网络采用简单的多层感知机（2 层，每层 512 个单元），即无需像 LSTM 网络那样具备记忆功能。Sievers 等人 [3] 表明，即使时间范围较短（0.5 秒），这样的网络也能学会复杂的手内操作任务。其优点是，该网络更易于训练，也更便于在预定义配置中进行研究。

III. 控制器的学习

控制器网络作为强化学习框架中的策略进行训练。我们采用离策略算法 Soft Actor-Critic（软演员评论家算法，SAC） [13]。我们自行实现的算法以及使用物理引擎 PyBullet 进行的仿真 [14]，已证明适用于现实世界机器人技术 [3]。

整体控制架构总结于图3中。请注意，在我们之前的工作中，我们通过采用较小的关节范围来限制手指的运动范围。现在，针对更复杂的目标旋转任务，我们采用完整的运动范围。我们发现，由于运动范围扩大，必须修改控制器，使策略 π 输出

表 I：观测

Name	Q函数	策略
关节角度	q_t	
期望角度	$\bar{q}_{t-1}$	
控制误差	$\bar{q}_{t-1} - q_t$	
目标姿态	R_{goal}	
立方体状态	$(x_t, R_{\text{sym}, t})$	$(\hat{x}_t, \hat{R}_{\text{sym}, t})$
旋转增量	$R_{t-1}^{-1} R_t$	R_t^{-1} 目标
立方体线速度	$v_t = \dot{x}_t$	

注：神经网络的完整输入由上方所示各量的长度为 $S = 5$ 的时间序列构成。为利用立方体的对称性，其姿态 R 通过八面体群约化为 R_{sym}。旋转以四元数形式输入。

相对于当前关节角度的角度（而不是直接输出预滤波的期望关节角度 $\tilde{q}$）。

$$\tilde{q}_{t+1} = \text{clip}(q_t + \pi(o_t)\frac{\tau_{\max}}{K_{\text{p}}}, q_{\min}, q_{\max}) \tag{1}$$

传递给策略的完整观测 o_t 如表 I 所示，并在第 III-C 节中说明。K_{p} 是底层阻抗控制器的常数，$\tau_{\max}$ 是允许的最大力矩。

此外，我们现在使用 120 个工作进程运行仿真来收集数据，而不是 12 个。增加工作进程数量有助于应对更广泛的域随机化。为了利用软演员评论家算法的离策略特性，我们还将回放缓冲区尺寸扩大了 10 倍，达到 150 万步。训练的所有细节，包括学习和仿真参数，均可在配套网站上找到。

A. 学习环境

虽然我们要解决的任务是将立方体重新定向到 24 种可能的 $\pi/2$ 目标姿态中的任意一种（参见图 6），但在训练期间，一旦达到当前目标，我们就会继续采样新的目标姿态。通过此任务进行学习，应能避免短视行为并增强对初始条件的鲁棒性，从而提升仿真到现实性能。

1）初始状态：通过为立方体设置随机姿态并将其放置在另一个固定立方体上方，重置环境。随后，手闭合，方式与在真实系统上执行的方式相同，然后移除固定立方体。再经过几个仿真步骤后，我们将此状态用作学习算法的首次观测。

2) 终止条件：如果立方体掉落（在 x_3 方向上到达 -5 cm）或移动到距离原点超过 10 cm 的位置，则回合以失败告终。若立方体在距离目标位置 $|x| < 2.5$cm 以内且方向变化量角度为 $\theta < 0.4$rad（参见图 2）的范围内保持 400 ms（4 个更新步骤），则视为成功达到目标。只有这些条件会向学习算法发出终止信号。此外，如果 10 s 后仍未达到目标，或总时间已过 120 s，我们会重置环境，但不发送终止信号，以确保回放缓冲区始终包含来自域随机化不同实例的数据。

B. 奖励

在训练过程中（参见第 III-E 节），我们使用两种不同的奖励函数。最初，我们采用 Chen 等人 [8]提出的“目标”奖励函数r_{g}。

$$r_{\text{g}} = \frac{\lambda_\theta}{\theta + \epsilon_\theta} - \text{clip}(\lambda_{\text{pos}}\|x\|^4, 0, \lambda_{\text{clip}}) + \begin{cases} \lambda_{\text{drop}}, & \text{if drop} \\ \lambda_{\text{succ}}, & \text{if succ.} \\ 0, & \text{else.} \end{cases}$$

第一项奖励较小的方向变化角度 θ，并加入一定的 ϵ_θ 以避免奇异点。第二项则惩罚立方体位置的偏差。是否达到目标或立方体是否掉落也被明确纳入其中。不同项由 λ 因子加权；在上下文明确时，我们省略时间索引 t。

随后，我们转而采用一种“更简单”的奖励函数 r_{s}，该函数基于位置 $\Delta x_t = \|x_t\| - \|x_{t-1}\|$ 的相对变化量和方向变化量 $\Delta\theta_t = \theta_t - \theta_{t-1}$。

$$r_{\text{s}} = \text{clip}(-\lambda'_\theta \Delta\theta, -\infty, \lambda'_{\text{clip}}) - \lambda'_{\text{pos}}\Delta x. \tag{2}$$

注意，我们不会明确地针对成功或掉落事件给予奖励。由于稠密奖励，成功达到目标对策略而言是有利的；而立方体掉落则会受到奖励中位置分量的惩罚。

这种更简单的奖励函数的主要优势在于，最终回合奖励具有更好的可解释性（因为物理量以线性方式计入），并且对每步最大奖励进行裁剪，会迫使策略优先在随机化领域之间实现泛化，而不是针对特定环境配置优化高奖励。未来，我们希望完全避免使用目标奖励。

C. 观测

得益于模块化学习架构，我们可以独立于估计器训练策略。然而，这也提高了建模传感器噪声的重要性。通常会使用非对称观测，以特权信息的形式向 Q函数（或价值函数） [3, 4, 10] 提供额外观测。在本工作中，我们将策略观测与 Q函数观测完全分离，从而可以向 Q函数传递真实状态，并向策略传递（可能极其）嘈杂的信号。此外，只有 Q函数能够接收立方体线速度。完整概览请参见表 I。

D. 域随机化

为成功实现仿真到现实迁移，我们在仿真过程中应用域随机化。最重要的参数请参见表 II，完整列表请见网站。

1）传感器噪声：每个步骤都会对传感器噪声进行采样，其他所有随机化都在新回合开始前进行。我们对关节角度和立方体状态施加高斯噪声。但请注意，我们只在策略输入中加入传感器噪声（参见第 III-C 节）；如果策略接收的是估计器预测的立方体状态（即表 III 和基准测试中的 S4、S5），则不会额外添加噪声。

表 II：域随机化

参数	分布[1]	备注[2]
	每步	
q [弧度]	$\mathcal{N}(0,\ 0.02)$	
x [米]	$\mathcal{N}(0,\ 0.01)$	S4、S5、B：关闭
R [弧度]	$\mathcal{N}(0, 0.2)$	S4、S5、B：关闭
	每个回合	
q_{off} [弧度]	$\mathcal{U}(-0.04, 0.04)$	
η_{lat}	$\mathcal{U}(0.81, 0.99)$	
η_{spin} [$\frac{\eta_{\text{lat}}}{\text{牛顿}}$]$_3$	$\text{Log}\mathcal{U}(2\times10^{-4}\ \ 2\times10^{-2})$	B：固定

2）控制器：我们使用粘滞动作 [15] 来模拟真实系统中的通信延迟。我们按照 Sievers 等人 [3]的识别结果，对 K_p、 K_d、寄生刚度和关节角偏移 q_{off} 进行采样。

3) 立方体： 我们对立方体的质量、尺寸和初始位姿进行采样。我们对立方体施加小幅随机力和力矩。

4) 摩擦： 随机化摩擦参数至关重要（手指连杆和立方体的横向 η_{lat} 及自旋η_{spin} 摩擦）。关于自旋摩擦的解释及其对策略性能的实证影响，请参见 V-E 节。

E. 训练流程与课程安排

实验章节和配套视频中使用的策略采用多步流程进行训练，概览及策略在基准测试（V-A 节）上的表现见表 III。最初（S1、S2），策略接收来自模拟器的真实立方体状态，并添加噪声；之后（S5），我们输入估计器当时预测的立方体状态（参见 III-F 节）。

在第一阶段（S1），我们针对目标奖励 r_{g} 进行训练，并采用课程安排，逐步增加重力，直至达到 g （这在手内操作中很常见 [3, 8]）。

每次训练新策略时，我们都会使用所有权重（以及优化器状态）来初始化一个新的回放缓冲区，并在略作修改的设置中继续学习。在 S2 中，我们更改奖励函数，并增大一阶低通滤波器的滤波器常数（以减缓动力学过程，从而更易于仿真，也更安全地在现实世界中执行）。在 S5 中，我们将策略输入切换为使用预测的立方体状态。

总训练时间约为两周，使用的机器最多配备 80 个核心。详情请见项目网站。

F. 估计器在环

给定一个在仿真中表现良好的策略和一个经过充分训练的估计器（参见第 IV 节），我们进行最后一轮训练，使策略能够适应使用估计状态。我们在仿真过程中运行估计器，并将估计的立方体状态（不添加噪声）传递给策略，而 Q 函数仍接收真实状态。此外，我们在奖励函数中添加一项，以激励策略

[1]Log$\mathcal{U}$ 表示对数均匀分布。[2]表示在某些训练阶段是否关闭或固定随机化，参见表 III。B 表示基准测试（V-A 节）。[3]根据我们的参数化方式，PyBullet 计算出指尖与立方体之间的有效自旋摩擦为 $\eta_{\text{spin}}\eta_{\text{lat}}/0.9$。

采取能够产生可预测状态转移的动作。为此，在“估计器”奖励 r_{e} 中，我们惩罚位置和姿态的估计误差。

$$r_{\text{e}} = r_{\text{s}} + \text{clip}(\lambda''_{\text{pos}}\|\hat{x} - x\|^2 + \lambda_\phi\phi^2, 0, \lambda''_{\text{clip}}) \qquad (3)$$

其中 $\phi = d(\hat{R},\ R)$（估计姿态与真实姿态之间的夹角）。如果 $\|\hat{x} - x\| >$ 1.5 cm 或$\phi >$ 0.8 rad，我们就终止回合，以确保回放缓冲区中不会出现这样的回合：例如，估计器 $\pi/2$ 出现偏差，导致操作虽然按预期进行，却累积了大量负奖励。

IV. 学习状态估计器

通过考虑手和立方体的状态，可以完整描述手物体系统。由于关节位置和速度可通过本体感知传感器直接测得，我们只需估计立方体的状态

$$s_t = (x_t, R_t, v_t, w_t)^T \in \mathbb{R}^3 \times \text{SO}(3) \times \mathbb{R}^3 \times \mathbb{R}^3 \qquad (4)$$

包括笛卡尔位置 x_t、姿态 R_t、横向速度 v_t和旋转速度 w_t。

仅根据触觉反馈估计系统状态尤其具有挑战性，因为无法获取有关被操作物体的全局信息。我们确定了两个主要的不确定性来源：

- 不确定的动力学源于手与物体之间基于接触的交互（例如摩擦效应）难以精确建模。
- 物体形状本身存在歧义，因此仅凭触觉信息无法唯一确定物体的位姿。

贝叶斯滤波方法可用于系统地处理这些不确定性；然而，存在不确定性且高度非线性的动力学使得分析或仿真的动力学模型无法使用。我们采用 D2P2F [9], ，这是一种可微粒子滤波器 [11, 12] 变体，已被证明适用于纯触觉状态估计。

粒子滤波器维护一组粒子 $\{(w^{(i)},\ s^{(i)})\}_{i=1,\ \ldots,\ N}$，每个粒子都带有权重和状态。可微粒子滤波器学习可应用于粒子的表达性函数，从而在贝叶斯滤波框架内实现预测和更新步骤。具体而言，D2P2F 学习一个生成式提议分布 $F_\varphi(\cdot|s^{(i)}_{t-1},\ z_t,\ u_t)$ 和一个更新模型$G_\varphi(s^{(i)}_{t-1},\ z_t,\ u_t)$，其中 $z_t = (q_t,\ \dot{q}_t)$ 是观测向量，由测得的关节角度和关节速度组成； u_t 是控制输入，由期望的关节角度 $\bar{q}_t$组成。估计器以 100 Hz 的频率更新（参见图3）。有关 D2P2F 的详细说明，请参阅 R¨ostel 等人 [9] 。

A. 训练

我们将预测状态 $\hat{s}_t$ 定义为粒子位置的加权平均、粒子旋转的中心对象，以及横向和旋转

表 III：训练序列

Step	训练	速率 b
S1	基于真实立方体状态，在奖励 r_{g} 上训练策略 π_0'。	0.68
S2	基于奖励 r_{s} 优化策略，得到 π_0.	0.99
S3	使用由 π_0 生成的数据训练估计器 f_0。	0.74
S4	基于来自 π_0 的数据，迭代改进估计器 f_i 以及来自 f_{i-1} 闭环中的立方体状态。	0.76
S5	在奖励 r_{e} 上改进策略，直至达到 π_1，并在闭环中使用 f_i。	0.92

速度的加权平均。我们通过最小化损失函数来优化 D2P2F 中 F_φ 和 G_φ 的权重

$$\mathcal{L}_\varphi = \frac{1}{T}\sum_{t=1}^{T}\sum_{j} c_j d_j(\hat{s}_{j,t}, s_{j,t})^2, \tag{5}$$

对于包含 T 个时间步的轨迹，其中 $d_j(\cdot)$ 是与 s的第 j个组件相关联的距离函数；对于笛卡尔维度，采用欧几里得距离；对于方向，则采用两个姿态之间的夹角。加权系数 c_j 用于补偿不同维度的尺度差异。我们设定 $c_x = 1.0$、$c_R = 100.0$ 和 $c_v = c_w = 0.1$。滤波器以监督方式在仿真中策略展开轨迹上进行训练。

训练分为三个阶段：首先，使用单个粒子训练提议模型，使其进行提前 1 步的预测（$T = 1$），直至收敛。然后，通过时间反向传播，在包含 $T = 100$ 个时间步的序列上训练滤波器。为此，从正态分布 $\mathcal{N}(s_0,\ \sigma)$ 中采样初始粒子，其中 σ 对应于训练数据的标准差，而 s_0 是在训练期间加上小偏置项后的真实初始状态。最后，在仿真中生成额外的训练数据，此时策略以当前模型的预测作为输入。将新收集的闭环数据追加到原始数据集中，并对模型训练 2 个轮次。之后，更新用于数据收集的滤波器模型。重复这一过程，直到闭环数据与离线数据样本的比例达到 1/2；此时我们发现，生成更多闭环样本并不会降低预测误差。

V. 实验

A. 基准测试协议

由于我们没有外部跟踪系统，因此无法在真实机器人上计算奖励。我们改为检查估计器与控制器的组合能否到达立方体的所有 24 $\pi/2$方向。

立方体由人以与仿真中初始化回合相同的方式递交。随后选定一个目标，机器人必须旋转立方体，使其与目标姿态一致。如果立方体掉落，则回合停止并判定为失败。

在仿真中对策略进行基准测试时，我们针对 24 个目标与 3 种自旋摩擦值的每种组合分别运行 8 次（$\eta_{\mathrm{spin}} = 2\times10^{-4}$、$1\times10^{-3}$、$1\times10^{-2}$）。我们固定自旋摩擦，因为它对结果影响很大。此外，我们将立方体尺寸固定为标称尺寸 8 cm，同样是因为其影响很大（而且真实立方体可以准确测量）。其他域随机化仍然启用。

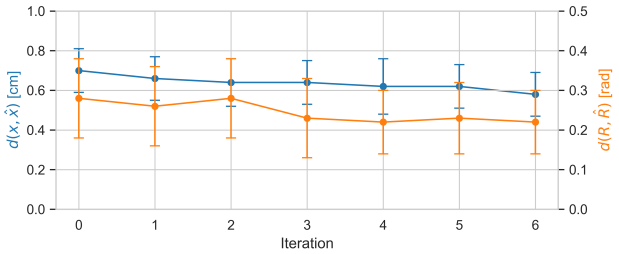

图 4：估计器与策略进行迭代训练期间的位置预测误差 $d(x,\ \hat{x})$ 和旋转预测误差 $d(R,\ \hat{R})$。

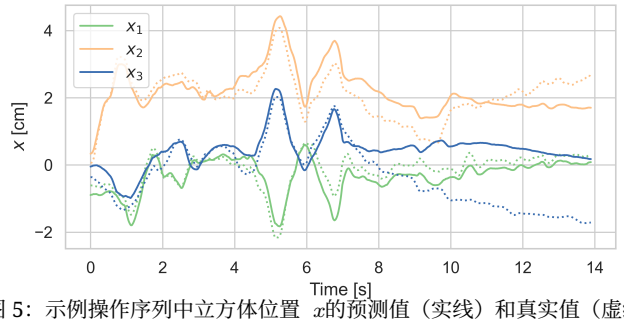

图 5：示例操作序列中立方体位置 x的预测值（实线）和真实值（虚线）。

B. 策略评估

表 III 详细列出了训练步骤，并报告了基准测试中的中间成功率。完成第一个步骤后，策略已经能够达到所有目标，但成功率还不够稳定。使用“简单”奖励函数进行微调后，即使在执行时使用带噪声的（模拟器）状态，策略的成功率也接近 100%。然而，将估计器纳入环路会大幅降低性能。迭代优化估计器可略微提升性能。最后，使用最新一轮估计器对策略进行微调，显著提升了基准测试表现。我们还在图 6 中展示了各个目标的成功率，以及真实机器人上的结果。在真实系统上，我们针对每个目标运行了 4 次，因此结果仅供定性参考。

C. 评估估计器

我们评估状态估计器在具有不同领域随机化参数的闭环展开过程中的预测准确度。如图 4 所示，尽管估计器在离线设置下生成的数据上训练至收敛，但在迭代训练过程中，预测误差仍在下降。这证实了使用与策略协同生成的数据训练估计器的重要性。不过请注意，预测准确度的提升仅使基准测试任务的性能略有改善（比较表 III 中的 S3 和 S4）。

在许多操作回合中，如图 5 所示，x_3 分量上的预测误差会随时间累积。这可以解释为，仅凭握住立方体的四个侧面，无法唯一确定其位置的高度。只有当手指触及立方体的上边缘（参见图 2）时，才能对x_3 位置作出判断。

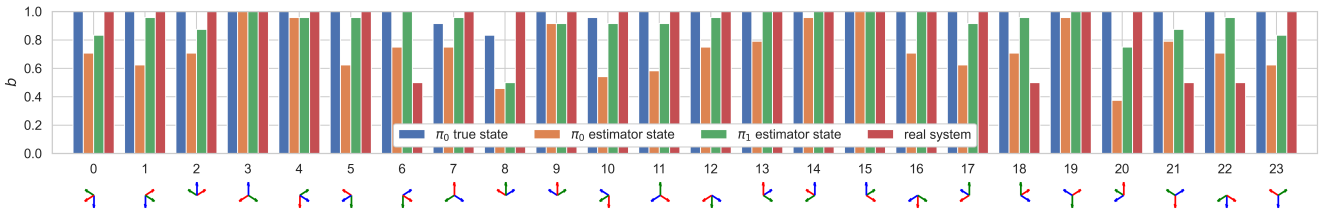

图 6：各目标姿态的平均成功率 b，并展示全部 24 种多重 $\pi/2$ 方向（八面体群）。请注意，目标 3 是立方体的初始姿态（即无需旋转）。测试协议的详细信息请参见 V-A 节。

D. 真实世界实验

我们将通过迭代训练过程获得的策略 π_1 应用于真实系统。请参阅随附视频或项目网站，观看机器人成功到达每个 24 $\pi/2$ 目标姿态（图 1 展示了一个示例）。此外，我们在图 6 中绘出了每个目标姿态经过 4 次试验得到的实验成功率。有趣的是，机器人能够非常可靠地到达 24 个目标姿态中的许多个，其中一些姿态在仿真中（涵盖整个域随机化范围）表现出的成功率较低。这意味着，对于这些运动，接触交互得到了良好建模，且处于控制器能够发挥作用的区域。另一方面，对于真实机器人而言，一些目标姿态（例如目标 6）要困难得多，而在仿真中几乎总能成功完成。这需要进一步分析，但我们推测，控制器利用了仿真中的某些交互，而这些交互并不符合现实或未被正确建模。

E. 自旋摩擦

对于仿真到现实性能而言，自旋摩擦尤其需要进行随机化。例如，当物体仅由两根手指夹持时，自旋摩擦的细微差异就可能导致截然不同的行为（物体卡住或向下摆动）。由于标称值很大程度上取决于具体的接触情况，因此难以确定。为此，我们在很宽的范围内随机化摩擦，使策略能够经历这些截然不同的行为，并找到鲁棒策略。

图 7 展示了让策略接触到如此宽范围的取值，会显著增加仿真中的问题难度。尽管绘图所用数据采用了与训练期间相同的采样策略收集，但策略在采样摩擦值较低的一端表现显著优于较高的一端。这表明，必须尽可能准确地识别并仿真系统，以避免策略无法在整个范围内泛化时，将重点放在不切实际的环境配置上。

F. 讨论

我们采用的模块化学习方法有一个关键优势：估计器的预测可以轻松检查和解读。在迭代训练流程的开发过程中，我们能够识别并解决估计器、策略以及各组件之间相互作用的不足。

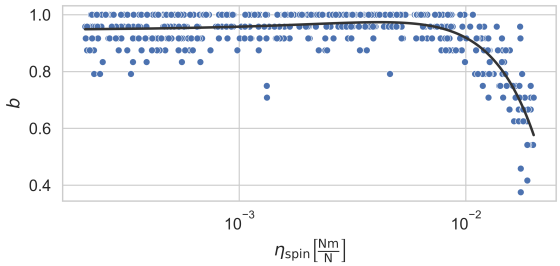

图 7：策略 π_1 在带有滤波器闭环的 24 个目标基准测试上的平均成功率，随自旋摩擦值范围 η_{spin}变化的曲线。

在推理过程中，我们可以检查预测的估计值（见图 1），从而便于在真实环境中调试模型。具体而言，我们发现这一点对于识别仿真与真实系统之间的差异至关重要。我们认为，V-A 节所述的基准测试设置尤其适合检测建模误差：灵巧操作任务要求精确控制手指的运动，而任务能否成功取决于手与物体之间接触诱导动力学的细微特性（见 V-E 节）。

VI. 结论

我们证明，纯触觉手内操作（手倒置）能够完成将立方体重新定向至目标姿态的任务。在仿真中，通过显著的域随机化，成功率达到 92%；通过向力矩控制的 DLR-Hand II 进行零样本仿真到现实迁移，全部 24 种目标姿态均能以较高成功率达到。为学习这项任务，我们提出了一种模块化深度强化学习架构，包含两个组件：用于控制手指的策略（其观测中明确包含立方体状态），以及用于根据实际和期望关节角度估计该状态的深度可微粒子滤波器（间接利用力矩测量值）。这种模块化设计支持高效的分步训练，并能深入了解各组件（尤其是滤波器具有可解释的隐藏状态，而非使用循环策略网络）。

未来，我们希望利用自适应域随机化来简化训练流程，并将估计器不确定性作为额外输入，使策略更加鲁棒。此外，我们希望将策略和滤波器的学习整合到端到端方案中，以进一步提升性能。

参考文献

[1] J. Butterfass、M. Grebenstein、H. Liu 和 G. Hirzinger，《DLR-Hand II：新一代灵巧机器人手》，载于《2001 年 ICRA：IEEE 国际机器人与自动化会议论文集》（目录号 01CH37164），第 1 卷，2001 年，第 109–114 页，第 1 卷。[2] B. B¨auml 等人，《Agile Justin：DLR 轻型力矩控制人形机器人家族的升级成员》，载于《IEEE 国际机器人与自动化会议论文集》，2014 年。[3] L. Sievers、J. Pitz 和 B. B¨auml，《使用力矩控制手进行纯触觉手内操作学习》，载于《2022 年国际机器人与自动化会议（ICRA 2022）》，美国宾夕法尼亚州费城，2022年5月23日至27日。IEEE，2022 年，第 2745–2751 页。 [在线]。网址：https://doi.org/10 1109 46639 2022 9812093 。 /ICRA . . [4] OpenAI，《利用机器人手进行灵巧的手内操作学习》，CoRR，2018 年。 [ ] 1808 00177 在线。网址：http://arxiv.org/abs/ 。A. Handa 等人，《Dextreme：将灵巧的手内操作从仿真迁移到现实》，arXiv，2022 年。[6] H. Qi 等人，《通过快速运动适应实现手内物体旋转》，载于《机器人学习会议（CoRL）》，2022 年。[7] G. Khandate、M. Haas-Heger 和 M. T. Ciocarlie，《利用本体感知学习手指步态式手内操作的可行性》，载于《2022 年国际机器人与自动化会议（ICRA 2022）》，美国宾夕法尼亚州费城，2022 年 5 月 23 日至 27 日。IEEE，2022 年，第 2752–2758 页。 [在线]。网址：https://doi.org/10 1109 46639 2022 9812212 。 /ICRA . . T. Chen、J. Xu 和 P. Agrawal，《通用手内物体重新定向系统》，载于《机器人学习会议论文集》，2021 年。[9] L. R¨ostel、L. Sievers、J. Pitz 和 B. B¨auml，《触觉手内操作状态估计器的学习》，载于《智能机器人与系统国际会议论文集》（已投稿，https://bit.ly/395h7ha），2022 年。[10] O. OpenAI 等人，《用机器人手还原魔方》，CoRR，第 abs/1910.07113 卷，2019 年。 [在线]。网址：http://arxiv.org/abs/ 1910.07113[11] R. Jonschkowski、D. Rastogi 和 O. Brock，《可微粒子滤波器：结合算法先验的端到端学习》，载于《机器人学：科学与系统论文集》，2018 年。[12] P. Karkus、D. Hsu 和 W. S. Lee，《粒子滤波网络及其在视觉定位中的应用》，载于《机器人学习会议》，2018 年。[13] T. Haarnoja 等人，《用于机器人操作的可组合深度强化学习》，CoRR，第 abs/1803.06773 卷，2018 年。 [在线]。网址：http://arxiv.org/abs/ 1803 06773 。E. Coumans 和 Y. Bai，[14]《PyBullet：用于游戏、机器人学和机器学习物理仿真的 Python 模块》，http://pybullet.org，2016–2021 年。[15] M. C. Machado 等人，《重新审视街机学习环境：通用智能体的评估协议与开放问题》，《人工智能研究杂志》，第 61 卷，第 523–562 页，2018 年。