

用于轮式移动机器人可靠操作的深度神经网络安全鲁棒自适应控制合成

Mehdi Heydari Shahna, and Jouni Mattila

Abstract—深度神经网络 (DNNs) 能够在避免动态建模的需求的同时实现精确控制并保持低计算成本。然而, 对于重型轮式移动机器人 (WMRs), 这些黑箱方法的部署仍然具有挑战性, 因为它们需要遵循严格的国际标准, 并且容易出现故障和干扰。我们设计了一种用于重型 WMR 的分层控制策略, 并由两个具有不同权威等级的安全层进行监控。为此, DNN 策略被训练并部署为主要控制策略, 在名义操作条件下提供高精度性能。当外部干扰出现并达到一定强度, 以至于系统性能低于预定阈值时, 低级安全层会介入, 通过停用主要控制策略并激活无模型鲁棒自适应控制 (RAC) 策略来进行响应。此转换允许系统在确保稳定性的同时继续运行, 通过有效管理系统稳健性和响应性之间的内在权衡。一种高级安全层在操作期间不断监控系统性能, 无论使用何种控制策略。仅当干扰变得非常严重以至于补偿再无效并且继续操作会危及系统或其环境时, 它才会启动关闭。所提出的 DNN 和 RAC 策略的合成在某种程度上保证了整个 WMR 系统的一致指数稳定性, 同时遵守安全标准。通过使用一台 6,000 公斤的 WMR 的实时实验, 进一步验证了所提出方法的有效性。

Note to Practitioners—在工业和建筑应用中, 减少劳动力成本和自动化复杂的重复性任务的需求日益增加, 这推动了从重型机械向大型机器人转型。因此, 预计配备复杂驱动机制的重型轮式移动机器人 (WMRs) 将在恶劣环境中运行, 并执行高功率的动态任务。深度神经网络 (DNNs) 可以通过消除在此类应用中对复杂建模的需求, 在部署时以低计算成本实现精确控制。然而, 确保可解释性和分析基于学习的黑箱非确定性控制策略的稳定性是一个主要挑战。这个挑战在重型轮式移动机器人中尤为明显, 因为它们易受故障和外部干扰的影响, 同时需要遵守如 ISO 3691、ISO 13849 和 IEC 61508 等严格的安全和可靠性标准。受 ISO/IEC TR 5469 第 10 节——人工智能——功能安全和 AI 系统的启发, 我们通过为配备主动悬挂转向架的重型 WMR 设计一种新颖的分层控制策略来应对这一挑战, 该转向架具有串并联结构的驱动链, 由两个具有不同权威级别的安全层监测, 而不需要了解执行器模型和车轮-地面相互作用。当 WMR 遇到严重外部干扰时, 低级安全层会停用初始高精度的基于 DNN 的控制策略, 激活监督鲁棒自适应控制 (RAC) 策略作为替代。这通过增强系统的鲁棒性来补偿干扰的影响, 并使其在操作期间的剩余时间内继续安全运行, 尽管以降低精度为代价。高级安全层在干扰下持续监控系统性能, 如果认为干扰补偿可行, 则允许继续操作; 否则, 为了防止对系统或其环境造成风险, 将触发关闭。有趣的是, 将提出的 RAC 与高精度的 DNN 控制策略进行合成, 提高了在提出的分层控制架构中 WMR 的可解释性、鲁棒性和稳定性。该方法可能为学习算法在重型机器人系统中的更广泛使用铺平道路, 同时仍能满足严格的国际标准。该方法通过对单独的基于 DNN 的策略、单独的 RAC 策略以及在配备五级驱动机制的 4,836 kg WMR 上的合成的逐步实验评估进一步验证。对比结果证明了所提出控制框架的有效性。

Index Terms—Heavy-duty robots, robust control, actuators, neural networks.

Funding for this research was provided by the Business Finland partnership project “Future All-Electric Rough Terrain Autonomous Mobile Manipulators” (Grant No. 2334/31/2022).

All authors are with the Faculty of Engineering and Natural Sciences, Tampere University, 33720 Tampere, Finland. (e-mail: mehdi.heydarishahna@tuni.fi; jouni.mattila@tuni.fi).

深度神经网络 轮式移动机器人 障碍李雅普诺夫函数 鲁棒自适应控制 多用途部署器 永磁同步电动机 列文伯格-马夸尔特 均方误差 双曲正切 每分钟转数 预设性能控制 滑移转向轮式移动机器人 每分钟转数 人工智能 MPD $i = L(left), R(right)$ 的侧面 泵排量 (m^3/rad) PMSM/泵的角速度 (m/rad) 液压马达的角速度 (m/rad) 液压马达的角速度 (m/rad) 车轮的半径 (m) 液压泵流量 (m^3/s) 液压马达排量 (m^3/rad) 车轮的线速度 (m/s) 齿轮减速 正的常数 PMSM/泵的转速 (rpm) 正的常数 外部扰动和建模误差 正的常数 轮子的期望线速度 (m/s) 车轮 (m/s) 的速度跟踪误差 隐藏层总数 索引当前隐藏层 第 l 层的权重矩阵 第 l 层的神经元数量 与层 l 相关联的偏置向量 第 l 层的预激活向量 层 l 的激活向量 在层 l 应用的激活函数 MPD 的 i 侧的批输入矩阵 训练样本数量 (或批大小) 一个全为 1 的列向量 均方误差损失函数 预测输出矩阵 目标 (真实) 输出的矩阵 目标输出 所有可训练参数的集合 弗罗贝尼乌斯范数 第 p 个训练样本的预测误差 可训练参数的总数 展平的参数向量 所有训练样本的堆叠误差向量 网络输出层的维度 堆叠误差向量的雅可比矩阵 叠加误差向量的第 k 个标量分量 对神经网络中的可训练参数进行索引 索引堆叠误差向量的标量 MSE 损失函数的 Hessian 矩阵 LM 优化中的阻尼参数 梯度下降解释中的步长 单位矩阵 优化过程中的迭代索引 候选新权重 LM 算法中的阻尼调整因子 LM 算法中的阻尼调整因子 PPC 对跟踪误差 e_i 的界限 在 PPC 框架中的允许超调 在 PPC 框架中的最终稳态误差界限 PPC 框架中的收敛速度界限 PPC 对跟踪误差 e_i 的限制 PPC 框架中的可允许超调 PPC 框架中的最终稳态误差界限 PPC 框架中的收敛速度界限 i -th 侧的自适应律参数 i -侧的自适应法误差 未知正常数 正常数 正数常量 i 边的正二次函数 未知的正界 任意正常数 未知正常数 未知的正界 n_{p_i} 的 RAC 策略命令 n_{p_i} 的 DNN 策略命令 锁存关断逻辑 锁存打开逻辑 低级安全层中的性能指标 高级安全层中的性能指标

I. 介绍

A. 背景和背景

他 机器人化的概念可以追溯到古代和文艺复兴时代对自运行机器的迷恋, 最终发展成为现代自主机器人 [1] 的愿景。术语“机器人”于 1920 年被创造, 反映了创造人工劳动者的愿望。到 20 世纪中叶, 工业机器人开始在工厂中执行重复性任务, 其目标是提高生产力、精度和安全性 [2]。如今, 机器人可以连续运行, 擅长处理危险或单调的任务, 降低错误率, 并使人类能够专注于更高层次的功能 [3]。机器人化也通过将传统的人类操作设备转换为自主轮式移动机器人 (WMRs) [4], [5], 革新了

轮式建筑机械。重型建筑机器在隧道、矿山和建筑工地等危险且结构不清的环境中执行高功率任务，这些环境中不稳定的地形、有限的能见度和物理危险对设备和操作员构成风险。由于空间和环境的危险，一些区域，如深坑或封闭隧道内部，特别不安全或人类难以接近 [6]。这些因素强调了需要智能的自主或半自主的控制系统以在高风险和难以到达的条件下确保安全同时保持操作效率 [7]。履带式车轮移动机器人 (SSWMRs) 被广泛应用于越野和建筑环境中，因为其机械结构消除了复杂的转向连杆，并允许在原地进行旋转转弯，基本实现了零转弯半径。这一设计赋予了 SSWMRs 在狭小空间和紧凑施工现场的卓越灵活性，同时在困难地形上提供了出色的机动性和多功能性，以及成本效益高、维护费用低的操作。此外，大多数重型 SSWMRs 采用车轮液压马达，因为其在承受恶劣操作环境时，具有优越的扭矩密度和机械稳健性。液压马达特别适合需要在低速下高力输出的移动应用，例如在越野或不平坦地形上的牵引和导航。为了为这些液压马达供电，现代电液架构越来越多地采用永磁同步电动机 (PMSMs)，因为与传统机器相比，它们具有高功率密度、精确可控性，最重要的是具有卓越的效率。它们在低速下提供平稳扭矩的能力尤其有利于在各种负载条件下调节液压压力和流量。

B. 相关工作、研究空白与研究动机

基于模型的 SSWMR (自平衡两轮机器人) 控制由于其在自主导航中的关键作用，已成为学术和工业研究的一个突出焦点。在控制理论中，基于模型的这一术语指的是在控制器设计中利用系统动态知识的程度。当有精确的模型时，基于模型的控制器可以提供高性能和高精度。因此，[8] 中的作者提出了一种结合神经动态优化的模型预测控制方案，以实现两轮 SSWMR 的精确轨迹跟踪。同样地，提出了一种在 [9] 中的线性二次调节器和在 [10] 中的虚拟力控制，这两者通过结合 SSWMR 中四轮驱动配置的动态模型实现，其中每一侧的车轮是机械耦合的。显然，许多先进的基于模型的控制策略严重依赖于系统精确数学模型的可用性——通常是以传递函数或状态空间表示的形式。与轻型机器人不同，重型 WMR 通常包含多级驱动机制，以根据控制命令将能量源的动力传递到地面接触的车轮，以产生运动 [11], [12]。驱动链的每一级都具有不同的动态特性，这些特性相互非线性地交互，并且通常具有强烈的运动学耦合。这些相互依赖导致复杂的输入输出关系，其中某一阶段的扰动或延迟在整个系统中传播并降低整体性能。这使得重型 WMR 的参数丰富的建模在分析上复杂且开发和验证上计算量大。结果，许多基于模型的控制方法忽略了车轮与地面相互作用 [13] 和其他难以表征的系统行为的效应，而是致力于通过自适应控制技术 (如在 [14]–[16] 中提出的技术) 来增强系统鲁棒性。尽管这些技术增加了鲁棒性，但由于干扰抑制和系统响应性之间的基本权衡，它们可能降低跟踪精度，其中改善一个通常以牺牲另一个为代价 [17], [18]。

为了解决这些挑战，深度神经网络 (DNNs) 可以通过其强大的逼近和泛化能力来帮助捕捉机器人动态行为，同时减少计算工作量 [19]。与传统的基于物理的模型不同，这些模型需要详细了解系统参数，DNNs 能够直接从传感数据中学习复杂的非线性关系，包括摩擦、轮地交互和执行器延迟等难以建模的现象 [20]。训练完成后，DNN 可以用轻量级推理过程替代或增强计算昂贵的动态方程，显著降

低实时计算负担。这使得 DNNs 对高维系统尤为有价值，例如配备了多级结构的主动悬挂转向架的重型 SSWMRs，其中传统的基于模型的控制分析上可能难以解决或计算上难以实现 [21], [22]。尽管 DNN 在其训练域内学习复杂动态模型时获得了高精度，当面临未知的外部力或在训练分布之外的操作条件时，基于 DNN 的控制策略本质上缺乏鲁棒性 [23], [24]。这种局限性主要来源于 DNNs 的数据驱动性质，它们仅能在其训练数据的统计边界内可靠地泛化。此外，收集能够捕获机器人可能遇到的所有操作条件的数据通常在时间和计算成本上是不切实际的——尤其是因为某些条件可能永远不会在实践中发生，可能很少发生，或者在数据收集过程中可能被忽视了 [25]。另外，与基于模型的控制器不同，基于学习的控制策略具有黑箱性质，使得它们的可解释性有限，形式稳定性分析极具挑战性 [26], [27]。因此，确保在不确定或安全关键条件下的可靠性能仍然是一个重大挑战 [11]。这种无法确保性能和操作可靠性的情况对于重型机械是不可接受的，重型机械必须遵循严格的国际安全和功能标准，例如 ISO 3691 [28]、ISO 13849 [29] 和 IEC 61508 [30]，以防止对环境和人员造成伤害。考虑到此类机器人系统预期在恶劣、无结构的环境中运行并执行高功率动态任务，这一问题尤为关键，这使得它们本质上容易受到外部干扰、故障和失效模式的影响。

针对上述挑战，本研究旨在解决一个根本问题：学习型控制策略如何能够有效地部署在配备复杂执行机构的重型轮式移动机器人上，从而在无需大量分析建模的情况下实现高性能，并在符合相关国际安全标准的条件下保证安全和系统稳定性？在正式的控制理论中，通常使用如屏障李雅普诺夫函数 (BLFs) [31], [32] 等技术来为动态和混合系统建立安全边界。这些方法确保系统保持在预定义的安全操作区域内。然而，得到的集合通常较为保守，因为一旦接近边界，这些方法并不会主动将系统驱动回更安全的状态 [33]。

C. 本文的贡献和结构

受 ISO/IEC TR 5469 第 10 节的启发 - 人工智能 (AI) 一功能安全和 AI 系统 [34]，监督屏障函数可以作为监控机制来检测因 AI 技术引起的潜在不安全行为，无论是由于内部故障还是外部干扰。一旦检测到不安全行为，系统可以启动传统的稳定控制器以维护安全性。受该标准的启发，我们提出了一种用于重型 WMR 的分层控制策略，由具有不同权威水平的两个安全层进行监控。为此，采用从车载传感器收集的数据来训练高性能 DNN 控制策略，捕捉实际运动和控制命令之间的关系。这些数据是通过简单的开环控制器生成的，在一个安全的环境中在人类监督下，逐步增加控制命令输入至最大值以记录相应的系统响应。然后，将训练好的 DNN 控制作为正常操作期间的初始控制策略进行部署，以提供高精度性能。如果发生外部干扰且强度足够大，导致控制性能未能满足规定的阈值，则低级安全层会停用基于 DNN 的控制策略，并启动无模型鲁棒自适应控制 (RAC) 策略作为替代，确保持续安全操作。通过管理系统鲁棒性和响应性之间的基本权衡，这种转变补偿了外部干扰的影响，确保了通过增强鲁棒性而降低系统响应性来实现剩余操作期间的持续安全运行。无论使用何种控制策略，都会采用对数 BLF 作为高层安全层，以在运行过程中持续监控系统性能。只有当干扰变得足够严重以至于补偿不再可行，并且继续运行会对系统或其环境构成风险时，才会触发关闭。本研究的关键发现总结如下：

1) 高精度跟踪性能通过 DNN 和安全 RAC 的综合实现, 同时根据 ISO/IEC TR 5469 确保对外部干扰的鲁棒性。这消除了对重型 SSWRM 固有复杂驱动机制的明确了解的需要

2) 控制跟踪误差阈值的要求, 如超调和稳态误差, 在两个不同层级安全层内是可调的。这种适应性使系统能够满足目标应用的具体性能要求。

3) 所提出的控制框架确保了整个 SSWRM 系统的统一指数稳定性, 即使在存在黑箱学习模型和外部干扰的情况下。

本文的其余结构如下。第 ?? 节详细描述了系统建模和问题表述。第 ?? 节介绍了一种特定于在标称条件下运行的 SSWRM 的基于 DNN 的控制策略设计, 利用已有的传感器数据记录。为了增强系统对外部干扰的鲁棒性, 第 II 节概述了所提出的无模型 RAC, 其中包括针对机器人每一侧的自适应法则。这两种控制策略都在第 ?? 节中详尽地开发在一个两层安全架构内, 每层在不同的权威级别下操作。在第 III 节中, 通过对独立的基于 DNN 的策略、独立的 RAC 策略及其在一个装备有五级驱动机构的 6,000 kg 的 WMR 上的综合进行逐步实验评估来进一步验证这一方法。比较结果展示了所有提出的方法在实际场景中的可行性。第 ?? 节对研究的关键发现进行了简明总结, 并给出本研究得出的总体结论。附录中提供了详细的 MATLAB 编码说明, 以促进复现和对已实现方法的深入理解。

在第 I-A 节之后, 考虑一个由 PMSM 驱动的重型 SSWMR, 该 SSWMR 配备有两个相同的多级驱动机构——每侧一个——将电能和液压能转换为机械运动, 如图 ?? 所示。

此类应用通常在每一侧都具有两个相同的主动悬挂转向架, 特征为与线性执行器集成的串并联驱动结构。每侧由 PMSM 驱动, 其功能是整个机制的主要力生成器, 并且可以作为 SSWMR 的控制输入, 具体要取决于 PMSM 配置, 可以通过其信号控制——主要是其转速 (rpm) 或输出扭矩 (Nm)。因此, PMSM 驱动一个液压泵, 该泵为两个串联的液压马达提供动力, 这些液压马达在机械上耦合并通常以相同的角速度运行。这些液压马达通过齿轮箱提供扭矩, 以驱动对应的机器人的车轮。安装在其中一个液压马达或直接在轮子上的速度传感器测量角速度, 并提供实时控制反馈, 如图 1 所示。假设主要控制目标是设计控制规律, 为 PMSMs 生成 rpm 命令, 使每一侧的车轮速度能够准确跟踪预期的速度曲线, 从而确保平台的稳定和精确的运动控制。令 $i = R, L$ 分别表示 SSWMR 的左侧和右侧。由于泵直接耦合到 PMSM 的轴上 (没有齿轮箱, 没有皮带驱动), 泵的流量通常与角电动机和液压泵的速度 $\omega_{e_i} = \omega_{p_i} : \mathbb{R}^+ \rightarrow \mathbb{R}$ 成正比, 如下所示

其中, $D_p \in \mathbb{R}$ 为泵的排量 (m^3/rad)。假设流体不可压缩且泄漏可以忽略不计, 则通过串联液压马达的流量等于泵的流量:

$$Q_i = D_m \cdot \omega_{h_i} \Rightarrow \omega_{h_i} = \frac{D_p}{D_m} \cdot \omega_{p_i} \quad (1)$$

其中 $D_m \in \mathbb{R}$ 是电机排量 (m^3/rad), $\omega_{h_i} : \mathbb{R}^+ \rightarrow \mathbb{R}$ 是两个液压马达的角速度。由于电机是串联和机械耦合的, 它们以相同的速度旋转。然后, 液压马达通过齿轮减速 $G \in \mathbb{R}^+$ 驱动车轮。因此, 得到每侧的角速度为

$$\omega_{w_i} = \frac{1}{G} \cdot \omega_{h_i} = \frac{D_p}{GD_m} \cdot \omega_{p_i} \quad (2)$$

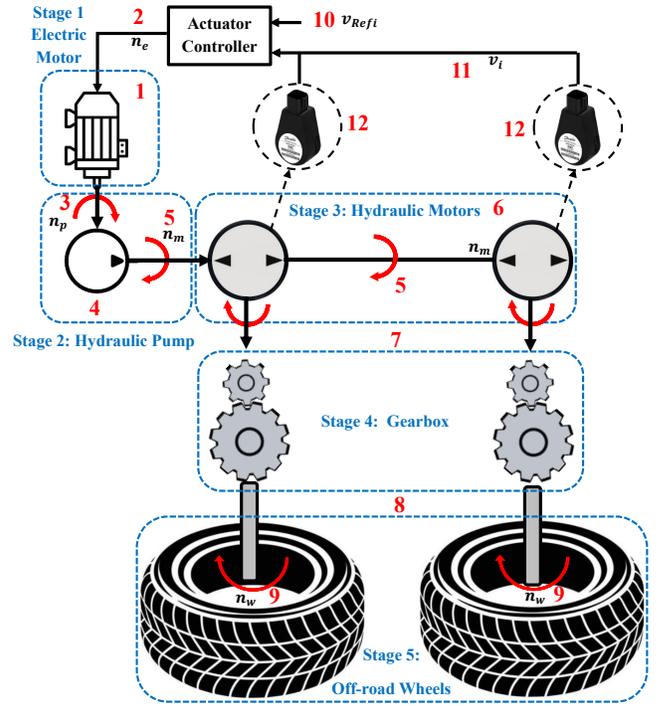


Fig. 1: 在 SSWMR 的每一侧都有相同的主动悬挂转向架。1 电动机驱动系统; 2 电动机的控制命令 (转/分钟); 3 液压泵与电动机耦合产生的速度 (转/分钟); 4 液压泵与电动机耦合以为液压系统供能; 5 液压马达产生的速度 (转/分钟); 6 串联液压系统 (两个相同的液压马达); 7 具有相同齿比的齿轮箱; 8 两个相同的车轮; 9 产生的车轮速度 (转/分钟); 10 参考车轮速度 (m/s); 11 测量的线速度 (m/s); 12 速度传感器

因此, 车轮的线速度为

$$v_i = r \cdot \omega_{w_i} = \frac{rD_p}{GD_m} \cdot \omega_{p_i} \quad (3)$$

其中 $r \in \mathbb{R}^+$ 是车轮半径 (假设车轮具有相同的半径)。由于泵/PMSM 的转速以 rpm 表示, 记为 $n_{p_i} = n_{e_i} : \mathbb{R}^+ \rightarrow \mathbb{R}$, 作为控制输入信号, 代入 $\omega_{p_i} = \frac{2\pi}{60} n_{p_i}$ 得

$$v_i = \frac{rD_p}{GD_m} \cdot \frac{2\pi}{60} \cdot n_{p_i} = K_v \cdot n_{p_i} \quad (4)$$

其中 $K_v = \frac{2\pi r D_p}{60 G D_m} \in \mathbb{R}^+$ 是一个正常数。方程 (4) 隐含地假设 PMSM 速度 n_{p_i} 的任何变化会立即导致车轮的线速度 $v_i : \mathbb{R}^+ \rightarrow \mathbb{R}$ 变化, 而没有考虑延迟或动态行为。然而, 在实践中, 涉及液压和机械组件的系统会表现出相当大的惯性和时间延迟, 这必须在实际建模和有效控制设计中加以考虑。添加动态最简单的方法是使用一阶微分方程, 如

$$\tau \frac{dv_i}{dt} + v_i = K_v \cdot n_{p_i} \quad (5)$$

这个动态方程在数学上相当于一个一阶低通滤波器, 用于平滑快速变化。方程 (5) 表明速度的变化率 $\frac{dv_i}{dt} : \mathbb{R}^+ \rightarrow \mathbb{R}$ 影响 v_i 响应速度。 $\tau \in \mathbb{R}^+$ 是一个正的时间常数, 决定响应是“慢”还是“快”。 $\tau > 0$ 意味着系统输出 v_i 落后于输入。

假设 II.1. 将 τ 设为零在物理上是不现实的, 因为所有的现实世界系统都表现出惯性和延迟, 阻止即时响应。

方程 (5) 仍然代表一个高度简化的模型，并忽略了显著影响系统动力学的几个关键因素。它忽视了系统不确定性、执行效率低下以及轮-地面相互作用和滑动等外部干扰。基于 [12]，我们假设一个对应外部干扰的未知非线性函数 $F_i(v_i, t) : \mathbb{R}^+ \times \mathbb{R} \rightarrow \mathbb{R}$ ，因此，

由于我们旨在设计一个无模型控制框架，我们假设所有的函数和系数 F_i 、 τ 和 K_v 是有界且未知的（参见假设 II.3）。因此，我们可以将方程 (??) 表示为

$$\dot{v}_i(t) = A_i n_{p_i}(t) + \tau^{-1}(F_i - v_i) \quad (6)$$

，其中 $A_i = \frac{K_v}{\tau} \in \mathbb{R}^+$ 。令 $v_{ref_i} : \mathbb{R}^+ \rightarrow \mathbb{R}$ 表示由运动规划部分根据应用目标生成的每一侧 SSWMR 的参考速度。如果我们定义每一侧的速度跟踪误差为 $e_i = v_i - v_{ref_i} : \mathbb{R}^+ \times \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$ ，基于方程 (6)，我们有

$$\dot{e}_i = A_i n_{p_i} + \tau^{-1}(F_i - v_i) - \dot{v}_{ref_i} \quad (7)$$

假设 II.2. SSWMR 每侧的目标加速度，用 \dot{v}_{ref_i} 表示，是有界的。此外，已知控制输入增益 A_i 为正，确保控制输入方向与 SSWMR 的运动方向一致。

假设 II.3. 我们注意到，由于 τ 和 v_i 的物理性质，它们本质上是有限的，而根据假设 II.2，假设 \dot{v}_{ref_i} 是有界的。与鲁棒控制文献中的既定结果一致（例如，[35]，[36]），我们进一步假设功能增益 A_i 和动态不确定性项 F_i 均被未知但有限的正常数所界定。这意味着对于所有可允许的 v_i 值和所有 t 而言， A_i 和 F_i 都不能无限增长。允许不受限制的不确定性将需要任意大的，可能无限的控制输入以确保鲁棒性和稳定性，这在现实系统中是一个不切实际且不安全的要求。这种过度的控制努力可能导致执行器饱和、机械故障或触发内置的紧急关闭协议。为解决这些实际约束，我们稍后将引入安全约束的控制设计策略，以明确考虑并遵循这些物理和操作限制。

备注 II.1. 在本节中，我们通过一个重型 PMSM 驱动 SSWMR 的实际例子展示了建模参数，以说明准确识别系统参数各阶段所带来的计算挑战。尽管存在这些复杂性，我们的目标是开发一个不依赖于任何详细系统建模的无模型控制框架，该框架不限于 PMSM 驱动的 SSWMR 应用。这种方法将系统视为一个黑箱——接受来自主要力发生器的速度或扭矩信号作为控制输入，并接收车轮运动反馈作为系统输出——从而消除了对内部动力学显式知识的需求。

鉴于 SSWMR 的驱动机制的每一侧的动态特性涉及单一输入和单一输出，我们为每一侧采用独立的前馈 DNN 策略。这些策略是使用从机载传感器获取的数据进行训练的，这些数据捕捉到驱动机制的非线性特性，即控制输入 (n_{p_i}) 与结果运动 (v_i) 之间的关系。为了收集这些数据，可以逐步将控制输入从零变化到其标称的正值和负值，同时从传感器记录对应的轮子速度。附录 A 提供了用于实现所提议 DNN 策略的核心 MATLAB 代码结构。

D. 架构和符号

前馈 DNN 架构由多层互连的计算单元（神经元）组成，每个单元应用线性变换，然后是非线性激活函数 [37]。在本研究中，SSWMR 每侧的线速度，用 v_i 表示，作为对应侧网络的唯一输入。每侧网络的输出是一个标量值，表示 PMSM 的转速，用 n_{p_i} 表示，作为控制命令。共有 $L \in \mathbb{R}^+$ 个隐藏层，其中第 ℓ 层隐藏层包含 n_ℓ 个神经元，

带有 $\ell = 1, 2, \dots, L$ 。输出层对应于第 $L+1$ 层，在这项工作中，由一个神经元组成。第 ℓ 层的权重矩阵表示为 $W^{(\ell)} \in \mathbb{R}^{n_\ell \times n_{\ell-1}}$ ，对应的偏置向量为 $b^{(\ell)} \in \mathbb{R}^{n_\ell}$ 。对于每层 ℓ ，预激活（仿射变换）表示为 $z^{(\ell)} \in \mathbb{R}^{n_\ell}$ ，激活后（即层的输出）表示为 $a^{(\ell)} \in \mathbb{R}^{n_\ell}$ 。第一层的输入表示为 $a^{(0)} := v_i \in \mathbb{R}$ 。对于每层 $\ell = 1, \dots, L+1$ ，预激活和激活的计算为

$$z^{(\ell)} = W^{(\ell)} a^{(\ell-1)} + b^{(\ell)}, \quad a^{(\ell)} = \varphi^{(\ell)}(z^{(\ell)}) \quad (8)$$

其中 $\varphi^{(\ell)}$ 表示第 ℓ 层的（逐分量）激活函数。对于隐藏层，我们采用双曲正切（Tanh）激活函数作为 $\varphi(z) = \tanh(z)$ ，其中 $\varphi(z) \in (-1, 1)$ [38]。用作输出层的回归（即线性激活）函数为 $\varphi^{(L+1)}(z^{(L+1)}) = z^{(L+1)}$ 。因此，网络的最终输出为 $n_{p_i} = a^{(L+1)} = z^{(L+1)} \in \mathbb{R}$ 。这使得网络能产生实值输出，适用于控制多阶段驱动系统。

给定输入标量 v_i ，网络输出是通过层级依次计算的。第一隐藏层的结果为

然后，后续的隐藏层 ($\ell = 2, \dots, L$) 被计算为最终，输出层 ($\ell = L+1$) 实现为当处理一批 P 样本时，输入被安排为矩阵的列形式：在矩阵形式中，层的输出被计算为

其中 $\mathbf{1} \in \mathbb{R}^P$ 是一列全为 1 的向量， $\varphi^{(\ell)}$ 对其参数逐元素应用。

E. 损失（成本）函数

由于我们使用回归任务，为了评估网络预测 n_{p_i} 与真实目标 t_{p_i} 之间的一致性，我们引入了一个基于均方误差 (MSE) 的损失函数 $E_{\text{MSE}}(\theta)$ ，其中 $\theta = \{W^{(\ell)}, b^{(\ell)}\}_{\ell=1}^{L+1}$ 表示所有权重和偏置的集合。 $E_{\text{MSE}}(\theta)$ 被定义为

$$E_{\text{MSE}}(\theta) = \frac{1}{P} \sum_{p=1}^P \left\| n_{p_i}^{(p)} - t_{p_i}^{(p)} \right\|_2^2 \quad (9)$$

同样地，如果我们将所有网络输出和目标堆叠成矩阵 $N_{p_i}, T_{p_i} \in \mathbb{R}^{n_{L+1} \times P}$ ，那么

其中 $\|\cdot\|_F$ 表示 Frobenius 范数。

虽然深度学习框架通常采用一阶（随机）优化算法，但中等规模的网络可以使用二阶启发的 Levenberg-Marquardt (LM) 反向传播方法进行训练。LM 将训练表述为一个非线性最小二乘问题，并通过 Gauss-Newton 方法加上阻尼参数来近似 Hessian。因此，它通常比传统的梯度下降收敛需要的迭代次数更少。对于具有参数向量 $w \in \mathbb{R}^{N_w}$ （堆叠所有权重和偏置）的网络，设

$$\xi_{p_i}^{(p)} = n_{p_i}^{(p)}(w) - t_{p_i}^{(p)} \in \mathbb{R}^{n_{L+1}} \quad (10)$$

为样本 p 的误差。注意，向量 $w \in \mathbb{R}^{N_w}$ 收集了网络中的每个可训练参数，即每个权重矩阵 $W^{(\ell)}$ 的所有条目和 $\ell = 1, \dots, L+1$ 的每个偏置向量 $b^{(\ell)}$ 的所有条目。具体来说，如果层 ℓ 有 $W^{(\ell)} \in \mathbb{R}^{n_\ell \times n_{\ell-1}}$ and $b^{(\ell)} \in \mathbb{R}^{n_\ell}$ ，那么 $W^{(\ell)}$ 的所有 $n_\ell \times n_{\ell-1}$ 个组件和 $b^{(\ell)}$ 的所有 n_ℓ 个组件被“展平”并连接成一个单一的长向量。总长度

$$N_w = \sum_{\ell=1}^{L+1} (n_\ell n_{\ell-1} + n_\ell) \quad (11)$$

简单来说是一个全连接神经网络各层的和，其中每一项表示 $W^{(\ell)}$ 中权重的数量加上 $b^{(\ell)}$ 中偏置的数量。然后训练

优化这个向量 w 以最小化选定的损失。将所有误差堆叠成一个向量得到

$$\xi_{p_i}(w) = \begin{bmatrix} \xi_{p_i}^{(1)} \\ \xi_{p_i}^{(2)} \\ \vdots \\ \xi_{p_i}^{(P)} \end{bmatrix} \in \mathbb{R}^{mP}, \quad m := n_{L+1} = 1 \quad (12)$$

。然后，总的 MSE（缩放比例为 $1/P$ ）为

$$E_{\text{MSE}}(w) = \frac{1}{P} \|\xi_{p_i}(w)\|_2^2 = \frac{1}{P} \xi_{p_i}(w)^T \xi_{p_i}(w) \quad (13)$$

。找到使 $E_{\text{MSE}}(w)$ 最小化的权重 w 是一个非线性最小二乘问题。

F. 雅可比和高斯-牛顿近似

雅可比矩阵 $J(w) \in \mathbb{R}^{(mP) \times N_w}$ 被定义为

$$J_{k,j}(w) = \frac{\partial s_k(w)}{\partial w_j} \quad (14)$$

其中 s_k 是堆叠误差向量 ξ_{p_i} 的第 k 个分量。换句话说， J 的每一行对应于 $\partial s_k / \partial w$ 。 E_{MSE} 的精确梯度为

$$\nabla E_{\text{MSE}}(w) = \frac{2}{P} J(w)^T \xi_{p_i}(w) \quad (15)$$

精确的海森矩阵定义为

$$H(w) = \nabla^2 E_{\text{MSE}}(w) = \frac{2}{P} \left(J(w)^T J(w) + \sum_{k=1}^{mP} s_k(w) \nabla^2 s_k(w) \right) \quad (16)$$

在高斯-牛顿方法中，忽略了第二项（涉及模型输出的二阶导数）。因此，

$$H(w) \approx \frac{2}{P} J(w)^T J(w) \quad (17)$$

$$\nabla E_{\text{MSE}}(w) = \frac{2}{P} J(w)^T \xi_{p_i}(w)$$

求解精确的牛顿步 $\Delta w = -H(w)^{-1} \nabla E(w)$ 代价很高（并且需要计算二阶导数）。高斯-牛顿通过 $J^T J$ 来近似 H ，得到一个步骤

$$\Delta w_{\text{GN}} = - (J^T J)^{-1} J^T \xi_{p_i} \quad (18)$$

。然而， $J^T J$ 仍然可能接近奇异或病态，导致更新不稳定。因此，LM 可以在 $J^T J$ 中添加一个阻尼 Tikhonov 正则化项 μI 。LM 步骤 Δw 求解

$$(J^T J + \mu I) \Delta w = -J^T \xi_{p_i} \quad (19)$$

。同样地，

$$\Delta w = - (J^T J + \mu I)^{-1} J^T \xi_{p_i} \quad (20)$$

。当 μ 很大时，项 μI 占主导地位，

$$\Delta w \approx -\frac{1}{\mu} J^T \xi_{p_i} = -\eta \nabla E_{\text{MSE}}, \quad \eta = \frac{1}{\mu} \quad (21)$$

，即一个小的梯度下降步骤。当 μ 很小 ($\mu \rightarrow 0$) 时，可得到高斯-牛顿步骤。因此，LM 自适应地在梯度下降（稳定但慢）和高斯-牛顿（快但可能不稳定）之间插值。为了

调整阻尼参数，在每次迭代 b 中，首先计算 $J_b = J(w_b)$ 和 $\xi_b = \xi(w_b)$ ；其次选择 μ_b （初始值较小）；第三，求解

$$(J_b^T J_b + \mu_b I) \Delta w_b = -J_b^T \xi_b \quad (22)$$

；第四，评估候选的新权重 $w_{\text{cand}} = w_b + \Delta w_b$ ；最后，计算新的误差 $E_{\text{MSE}}(w_{\text{cand}})$ 。如果 $E_{\text{MSE}}(w_{\text{cand}}) < E_{\text{MSE}}(w_b)$ ，则接受步骤 $w_{b+1} \leftarrow w_{\text{cand}}$ 并减少 $\mu_{b+1} = \mu_b / \beta$ （通常为 $\beta = 10$ 或类似因子）。如果 $E_{\text{MSE}}(w_{\text{cand}}) \geq E(w_b)$ ，则拒绝步骤 $w_{b+1} = w_b$ ，并增加 $\mu_{b+1} = \mu_b \times \beta$ 。通过在误差减少时减小 μ ，LM 接近纯高斯-牛顿，加速收敛到局部最小值附近。通过在步骤失败时增加 μ ，它表现得更像梯度下降，从而确保稳定性。为了说明用于 SSWMR 的基于 DNN 的控制策略的 LM 训练，给出了算法 1 的结构。

Algorithm 1: LM training for DNN-based control policy

Input : Initial weights w_0 , damping factor μ_0 , multiplier $\beta > 1$, tolerance ϵ , training data $\{(v_i^{(p)}, t_{p_i}^{(p)})\}_{p=1}^P$
Output : Trained weights w

```

1 Set  $w \leftarrow w_0$ ,  $\mu \leftarrow \mu_0$ 
2 repeat
3   Compute network outputs  $\hat{n}_{p_i}^{(p)}$  and errors  $\xi^{(p)} = \hat{n}_{p_i}^{(p)} - t_{p_i}^{(p)}$ 
4   Stack all errors into vector  $\xi(w) \in \mathbb{R}^P$ 
5   Compute Jacobian matrix  $J(w) \in \mathbb{R}^{P \times N_w}$ 
6   Solve:  $(J^T J + \mu I) \Delta w = -J^T \xi(w)$ 
7   Compute candidate:  $w_{\text{cand}} = w + \Delta w$ 
8   Evaluate  $E_{\text{cand}} = E_{\text{MSE}}(w_{\text{cand}})$ 
9   if  $E_{\text{cand}} < E_{\text{MSE}}(w)$  then
10    Accept step:  $w \leftarrow w_{\text{cand}}$ 
11    Decrease damping:  $\mu \leftarrow \mu / \beta$ 
12  else
13    Reject step:  $w$  unchanged
14    Increase damping:  $\mu \leftarrow \mu \cdot \beta$ 
15  until  $\|\Delta w\| < \epsilon$  or max iterations reached
16  return  $w$ 

```

- The Jacobian $J(w)$ contains partial derivatives of output error with respect to weights.

- All operations are performed on mini-batches or full datasets depending on the training set.

完成训练后，所得的模型参数封装了一种数据驱动的控制策略，在标称条件下能够高保真地将 SSWMR 的参考轮速 v_{ref_i} 映射为 PMSM 控制指令 n_{p_i} 。注意，DNN 使用实际系统输出数据 v_i 进行训练，而在运行中，理想速度参考 v_{ref_i} 传递给封装的基于 DNN 的控制器，以产生适当的控制指令。为了确保具有所提议的 LLM 训练的基于 DNN 的控制策略的 SSWMR 系统的高水平安全性，我们采用 PPC 框架来监测和调节系统响应的关键指标，如超调和稳态误差。在 PPC 中，通过战略性地使用奇点和零点等关键元素施加约束，维持系统的状态在期望的性能范围内。这通常是通过在控制律内嵌入障碍函数或转换来实现的，能够有效防止状态达到或超过不希望的限制 [31], [39]。根据 [35], [40] 中的方法，使用基于 DNN 的控制器的人每一侧的跟踪误差度量可以被限制在由 $-\zeta(t) < e_i < \zeta(t)$ 定义的指数、对称、随时间变化的区间内，其中：

$$\zeta(t) = (\zeta^{\text{shoot}} - \zeta^{\text{bound}}) e^{-\zeta^* t} + \zeta^{\text{bound}} \quad (23)$$

给定 $\zeta^{\text{shoot}} > \zeta^{\text{bound}} > 0$ 和 $\zeta_i^* > 0$ ， e_i 的超调被限制在 ζ^{shoot} 以下。参数 ζ^{bound} 和 ζ_i^* 分别定义了 e_i 的稳态界限和收敛速率。让我们为 SSWMR 的每一侧基于表达式

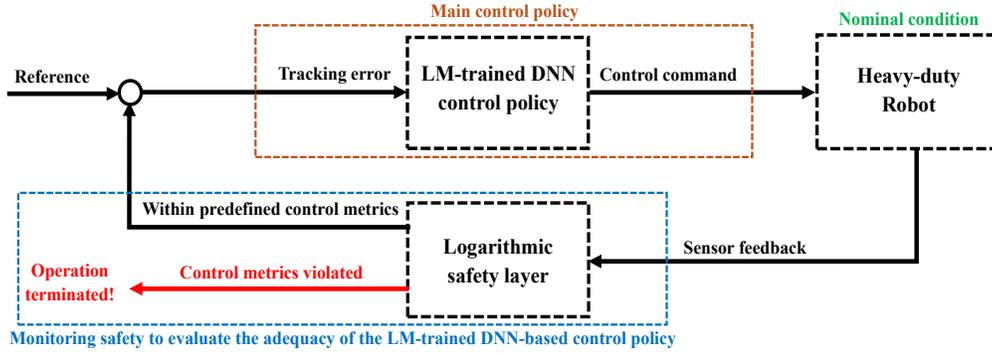


Fig. 2: 在标称条件下，提出的安全 LM 训练的基于 DNN 的 SSWMR 控制策略的示意图。

$\log\left(\frac{\zeta^2}{\zeta^2 - e_i^2}\right)$ 定义一个条件安全监控函数，其定义域受到分母保持正值的要求限制。如果初始误差满足 $e_i(t_0) < \zeta(t_0)$ ，那么当 e_i 趋近于 ζ 时，分母趋近于零，导致对数项发散到无穷大。这造成了一个奇点，使得函数由于对数内出现非正数参数而未定义。在实时环境中，这种条件安全监控机制可以很容易地实现，当检测到警告如“遇到无穷或 NaN 值”时可以停止执行，因为计算值超过数值限制。这一行为对确保跟踪误差不超过规定的界限 ζ 是有利的，从而防止在控制环境中出现潜在的不安全或不理想的系统响应。此条件安全监控机制的相关 MATLAB 代码在附录 B 中提供。

图 2 展示了所提议的采用 LM 训练的基于 DNN 的控制策略的标准示意图，该策略基于跟踪误差为重型机器人生成控制命令。为了确保安全，日志监控层评估控制性能是否保持在 PPC 范围内，如果检测到违规则停止操作。尽管安全层确保系统保持在预定义的安全操作区域内，但它起到被动作用——在边界违反时终止操作——而不是在接近界限时主动引导系统回到更安全的状态，导致结果安全集的保守性。为了说明用于 SSWMR 的 LM 训练的 DNN 控制策略，提出了算法 2 的结构。

Algorithm 2 : Safe LM-trained DNN policy for each side of the SSWMR

```

Input : PPC metrics for both safety layers  $\zeta^{\text{shoot}}$ ,  $\zeta^{\text{bound}}$ ,  $\zeta^*$ , reference velocities  $v_{\text{ref}_i}$ , sensor feedback  $v_i$ 
Output : Control command  $n_{p_i}$ 

1 for each side  $i$  do
2   Compute tracking error:  $e_i = v_i - v_{\text{ref}_i}$ 
3   Compute low-level safety:  $\zeta = (\zeta^{\text{shoot}} - \zeta^{\text{bound}}) e^{-\zeta^* t} + \zeta^{\text{bound}}$ 
4   Compute denominator:  $\text{denom} = \zeta^2 - e_i^2$ 
5   if  $\text{denom} \leq 0$  then
6     Raise safety violation and go to step 11
7   else
8     Pass  $v_{\text{ref}_i}$  to LM-trained DNN policy to compute  $n_{p_i}$ 
9     Return to step 3
10  end if
11  terminate operation
12 end for

```

- The PPC metrics are manually adjustable for use in a wide range of application scenarios.
- If safety condition is violated, n_{p_i} is not generated to protect the system

II. 安全的无模型 RAC 策略

在本节中，我们为研究中的 SSWMR 系统制定了一种稳定性保证的 RAC 策略。在控制理论中，鲁棒性和响应性之间存在一个基本的权衡——改善其中之一通常会降低另一个。在存在外部干扰的情况下，这种平衡尤为重要，因为增强的鲁棒性可以确保持续的安全运行。意识到这种权衡，我们为系统的每一侧提出了一种新颖的自适应控制策略，专门设计以提高鲁棒性而不影响整体稳定性。在 PPC 中，奇点和零点等关键元素被策略性地应用以施加约束，确保系统状态保持在期望的性能范围内。遵循 [41] 中的方法，我们为每一侧定义自适应律，如下

$$\dot{\hat{\theta}}_i = -\delta_i \hat{\theta}_i + \gamma_i \left(\frac{e_i}{o^2 - e_i^2} \right)^2 \quad (24)$$

其中 $\hat{\theta}_i(t_0)$ 、 δ_i 和 γ_i 为正常数。因此，机器人的每一侧的跟踪误差被限制在一个由 $-o(t) < e_i < o(t)$ 定义的对称、时变区间内。类似于方程 (23)，我们可以为控制误差性能定义一个衰减指数界限，如

$$o(t) = (o^{\text{shoot}} - o^{\text{bound}}) e^{-o^* t} + o^{\text{bound}} \quad (25)$$

其中 $o^{\text{shoot}} > o^{\text{bound}} > 0$ 和 $o^* > 0$ 。这种公式确保初始误差不超过 o^{shoot} ，而长期跟踪性能被 o^{bound} 限制，其收敛速率由 o^* 控制。假设自适应参数 $\hat{\theta}_i$ 估计参数 θ_i^* 。因此，我们可以定义自适应误差 $\tilde{\theta}_i = \hat{\theta}_i - \theta_i^*$ 。从方程 (24) 中，我们有

$$\dot{\tilde{\theta}}_i = -\delta_i \tilde{\theta}_i + \gamma_i \left(\frac{e_i}{o_i^2 - e_i^2} \right)^2 - \delta_i \theta_i^* \quad (26)$$

我们可以定义一个对数平方函数，如

$$V_i = \frac{1}{2} \log \left(\frac{o^2}{o^2 - e_i^2} \right) + \frac{1}{2} A_i \tilde{\theta}_i^2 \quad (27)$$

这个对数障碍函数作为安全度量，并可以如附录 D 所示在 MATLAB 中实现。对 (27) 微分后，我们有

$$\dot{V}_i = \frac{-\dot{o}e_i^2 + e_i o \dot{e}_i}{o(o^2 - e_i^2)} + A_i \tilde{\theta}_i \dot{\tilde{\theta}}_i \quad (28)$$

通过将 (7) 代入 (28)，我们得到

$$\begin{aligned} \dot{V}_i = & \frac{-\dot{o}e_i^2}{o(o^2 - e_i^2)} \\ & + \frac{e_i}{o^2 - e_i^2} [A_i n_{p_i} + \tau^{-1}(F_i - v_i) - \dot{v}_{\text{ref}_i}] \\ & + A_i \tilde{\theta}_i \dot{\tilde{\theta}}_i \end{aligned} \quad (29)$$

我们可以得到

$$\begin{aligned} \dot{V}_i = & \frac{-\dot{o}_i e_i^2}{o(o^2 - e_i^2)} + A_i \frac{e_i}{o^2 - e_i^2} n_{p_i} \\ & + \frac{e_i}{o^2 - e_i^2} \tau^{-1} (F_i - v_i) - \frac{e_i}{o_i^2 - e_i^2} \dot{v}_{ref_i} + A_i \tilde{\theta}_i \dot{\tilde{\theta}}_i \end{aligned} \quad (30)$$

现在, 我们为 PMSM 控制指令 n_{p_i} 提出 RAC 策略, 如将 (??) 代入 (30), 我们得到

$$\begin{aligned} \dot{V}_i \leq & A_i \frac{e_i}{o^2 - e_i^2} \left(-\frac{1}{2} k_i e_i - \gamma_i \frac{e_i}{o^2 - e_i^2} \hat{\theta}_i \right) \\ & + \left| \frac{e_i}{o^2 - e_i^2} \right| \left| \tau^{-1} (F_i - v_i) - \dot{v}_{ref_i} - \frac{\dot{o} e_i}{o} \right| \\ & + A_i \tilde{\theta}_i \dot{\tilde{\theta}}_i \end{aligned} \quad (31)$$

考虑在 (26) 中定义的提出的自适应法律误差, 我们得到

$$\begin{aligned} \dot{V}_i \leq & -\frac{1}{2} A_i k_i \frac{e_i^2}{o^2 - e_i^2} - A_i \gamma_i \left(\frac{e_i}{o^2 - e_i^2} \right)^2 \hat{\theta}_i \\ & + \left| \frac{e_i}{o^2 - e_i^2} \right| \left| \tau^{-1} (F_i - v_i) - \dot{v}_{ref_i} - \frac{\dot{o} e_i}{o} \right| \\ & - \delta_i A_i \tilde{\theta}_i^2 + \gamma_i A_i \left(\frac{\bar{e}_i}{o^2 - e_i^2} \right)^2 \tilde{\theta}_i - A_i \delta_i \theta_i^* \tilde{\theta}_i \end{aligned} \quad (32)$$

基于假设 II.3, 我们可以假设存在一个正参数 f_i^* , 使得

$$\left| \tau^{-1} (F_i - v_i) - \dot{v}_{ref_i} - \frac{\dot{o} e_i}{o} \right| \leq f_i^* \quad (33)$$

注 IV.1. 通过应用柯西-施瓦茨不等式, 可以证明对于任何标量 a 和 b , 积满足 $ab \leq \epsilon_i a^2 + \frac{1}{4\epsilon_i} b^2$, 其中 $\epsilon_i > 0$ 是任意正常数。

因此, 从备注 IV.1、(32) 和 (33), 以及已知 $\tilde{\theta}_i = \hat{\theta}_i - \theta_i^*$, 我们有

$$\begin{aligned} \dot{V}_i \leq & -\frac{1}{2} A_i k_i \frac{e_i^2}{o^2 - e_i^2} - A_i \gamma_i \left(\frac{e_i}{o^2 - e_i^2} \right)^2 \theta_i^* \\ & + \epsilon_i \left(\frac{e_i}{o^2 - e_i^2} \right)^2 f_i^{*2} + \frac{1}{4\epsilon_i} - \delta_i A_i \tilde{\theta}_i^2 - A_i \delta_i \theta_i^* \tilde{\theta}_i \end{aligned} \quad (34)$$

。让我们定义未知的正 θ_i^* 为

$$\theta_i^* = \frac{\epsilon_i f_i^{*2}}{\gamma_i A_i} \quad (35)$$

。因此, 我们可以达到

$$\begin{aligned} \dot{V}_i \leq & -\frac{1}{2} A_i k_i \frac{e_i^2}{o^2 - e_i^2} + \frac{1}{4\epsilon_i} \\ & - \delta_i A_i \tilde{\theta}_i^2 - A_i \delta_i \theta_i^* \tilde{\theta}_i \end{aligned} \quad (36)$$

。因此, 从引理 IV.1 和 (36), 我们有

$$\begin{aligned} \dot{V}_i \leq & -\frac{1}{2} A_i k_i \log \left(\frac{o_i^2}{o^2 - e_i^2} \right) + \frac{1}{4\epsilon_i} - \delta_i A_i \tilde{\theta}_i^2 \\ & - A_i \delta_i \theta_i^* \tilde{\theta}_i \end{aligned} \quad (37)$$

。显然,

$$\begin{aligned} \dot{V}_i \leq & -\frac{1}{2} A_i k_i \log \left(\frac{o_i^2}{o^2 - e_i^2} \right) + \frac{1}{4\epsilon_i} - \frac{1}{2} \delta_i A_i \tilde{\theta}_i^2 \\ & - \frac{1}{2} A_i \delta_i (\hat{\theta}_i - \theta_i^*)^2 - A_i \delta_i \theta_i^* (\hat{\theta}_i - \theta_i^*) \end{aligned} \quad (38)$$

。经过简化, 我们有

$$\begin{aligned} \dot{V}_i \leq & -\frac{1}{2} A_i k_i \log \left(\frac{o_i^2}{o^2 - e_i^2} \right) + \frac{1}{4\epsilon_i} - \frac{1}{2} \delta_i A_i \tilde{\theta}_i^2 \\ & + \frac{1}{2} A_i \delta_i \theta_i^{*2} \end{aligned} \quad (39)$$

, 我们有

$$\dot{V}_i \leq -\mu_i V_i + \ell_i \quad (40)$$

, 其中 $\mu_i = \min[A_i k_i, \delta_i]$ 和 $\ell_i = \frac{1}{4\epsilon_i} + \frac{1}{2} A_i \delta_i \theta_i^{*2}$ 。因此, 根据结果方程 (40) 和 [12], [17], [42]-[45], 即使在外部干扰存在的情况下, 公式 (7) 中提供的 SSWRM 系统的建议安全 RAC 策略依然是统一指数稳定的。

备注 IV.2. 参数 ℓ_i 在塑造系统的稳健性和响应能力方面起着关键作用。具体来说, ℓ_i 直接与 θ_i^* 相关, 后者描述了在方程 (33) 和 (35) 中外部扰动 f_i^* 的强度。如方程 (40) 所示, 如果 ℓ_i 过大, 则系统稳定性无法得到保证。这会违反 (25) 中定义的条件, 导致李雅普诺夫函数的导数为正, 从而表明不稳定性。当扰动 f_i^* 加剧时, 李雅普诺夫函数值趋近于零。如果扰动超过补偿控制器 (RAC) 根据致动器限制预定的补偿能力, 系统的稳健性阈值将被超越, 从而触发安全层进行关闭。为了展示 SSWMR 的安全 RAC 策略, 给出了算法 3 的结构。

Algorithm 3 : Safe RAC policy for each side of the SSWMR

Input : PPC metrics o^{shoot} , o^{bound} , o^* , reference velocities v_{ref_i} , sensor feedback v_i , design parameters δ_i , γ_i , k_i
Output : Control input commands n_{p_i}

```

1 for each side  $i$  do
2   Compute tracking error:  $e_i = v_i - v_{ref_i}$ 
3   Compute safety bound:  $o = (o^{\text{shoot}} - o^{\text{bound}}) e^{-o^* t} + o^{\text{bound}}$ 
4   Compute denominator:  $\text{denom} = o^2 - e_i^2$ 
5   if  $\text{denom} \leq 0$  then
6     Raise safety violation and go to step 12
7   else
8     Compute safety metric:  $\log \left( \frac{o_i^2}{o^2 - e_i^2} \right)$ 
9     Update adaptive parameter:  $\dot{\hat{\theta}}_i = -\delta_i \hat{\theta}_i + \gamma_i \left( \frac{e_i}{o^2 - e_i^2} \right)^2$ 
10    Compute control input:  $n_{p_i} = -\frac{1}{2} k_i e_i - \gamma_i \frac{e_i}{o(t)^2 - e_i^2} \hat{\theta}_i$ 
11  end if
12  terminate operation
13  end for
```

- The PPC metrics are manually adjustable for use in a wide range of application scenarios

- If safety condition is violated, n_{p_i} is not generated to protect the system.

- Adaptive parameter $\hat{\theta}_i$ is updated using a logarithmic Lyapunov-based law.

关于这种条件安全监控机制的相关 MATLAB 代码在附录 C 中提供。图 3 展示了所提出的稳定性保证 RAC 策略的标准示意图, 该策略基于跟踪误差为重型机器人生成控制命令。为了确保安全, 日志监控层在操作期间持续监控系统性能。仅当干扰变得足够严重, 以至于补偿不再可行, 且继续操作会危及系统或其环境时, 它才会启动关闭程序。

如前所述, 在实际应用中, 具有复杂驱动机制的重型轮式移动机器人主要在标称条件下运行, 此时所提出的 LM 训练的 DNN 控制策略通常可以实现高性能。这些控制器可能会在遇到轻微或短暂的外部干扰时不出现显著性能下

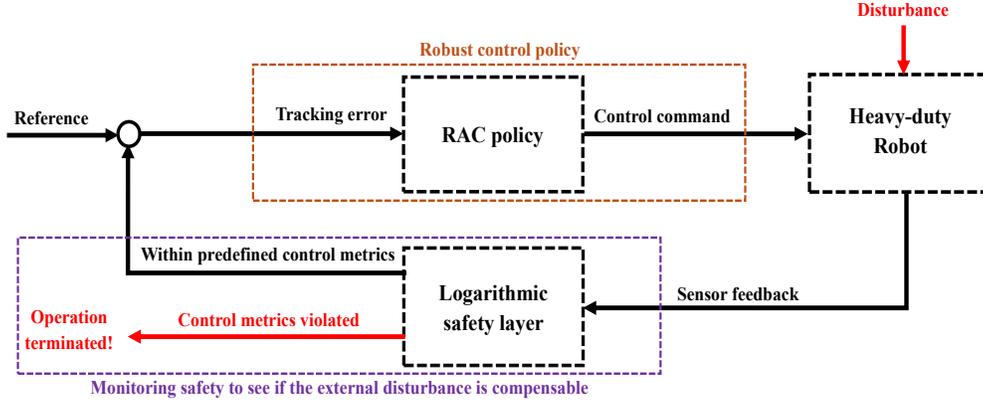


Fig. 3: 在扰动下，所提出的保证安全稳定的 SSWMR 自适应增益控制策略的示意图。

降。然而，为确保安全可靠的运行——特别是在涉及严重干扰或故障的罕见但关键的情景中——必须根据严格的国际安全标准来保证鲁棒性和系统稳定性。在本节中，我们提出将第 ?? 和 II 节中介绍的两种控制策略进行整合，并通过增加两个具有不同权限级别的安全层，以确保即使在可以补偿的外部干扰存在的情况下也能持续运行。该框架主要通过 LM 训练的 DNN 控制策略运行，该策略在标称条件下提供高跟踪精度，但鲁棒性有限。在正常运行期间，该策略确保优异的性能。然而，它的输出由低层安全层实时持续监控。如果外部扰动导致性能下降超过预定义的阈值，系统不会立即终止操作。相反，会激活一个监督 RAC 模块，使系统能够补偿扰动并继续安全运行，但在剩余的会话中跟踪精度将降低。通过这种方式，低层安全层管理稳健性和响应性之间的固有权衡。尽管 RAC 策略提供的精度低于 DNN 控制器，但其稳健性显著更高，使得系统能够可靠运行，直到规划的维护或纠正行动。值得注意的是，即使在 RAC 控制下，系统的行为仍继续由相同的安全层监控。如果扰动超过 RAC 的补偿能力——根据执行器的限制定义——则认为系统的稳健性阈值被违反，从而促使安全层启动受控关闭。这防止了系统、环境或周围资产的潜在损害。为实现这一安全机制，我们采用了一种锁存功能——一个有状态的构造，通过在触发条件满足时改变其输出来不可逆转地管理系统稳健性和响应性之间的固有权衡。其定义如下：

$$u_{c_i} = \alpha_1(t)u_{DNN_i} + \alpha_2(t)u_{s_i} \quad (41)$$

，其中 u_{s_i} 和 u_{DNN_i} 是针对 n_{p_i} 的 RAC 和 DNN 控制策略命令。采用锁存关闭逻辑，我们使用 $\alpha_1(t)$ 作为提出的基于 DNN 的控制，初始为 1，当条件满足（任何外部扰动发生并使性能降低超过预定义阈值时）时切换为 0，然后永久保持关闭。 $\alpha_1(t)$ 定义为如果我们将低级安全层边界定义为 $R = \zeta^2 - e_i^2$ ，我们可以在 MATLAB 中编写 α_1 ，如附录 E 所示。相比之下，我们在 DNN 控制不够稳健以维持跟踪性能的条件下，也使用了为建议的 RAC 策略的 α_2 。 α_2 首先启动锁存开启逻辑功能（从 0 开始），并在条件 ($R \geq 0$) 出现的时刻切换为 1，无论将来的输入如何，都保持在 1。 α_2 定义为

$$\alpha_2(t) = \begin{cases} 0, & \text{if } e_i^2(\tau) < \zeta^2(\tau) \text{ for all } \tau \in [0, t] \\ 1, & \text{if there exists } \tau \in [0, t] \text{ such } \zeta^2(\tau) \leq e_i^2(\tau) \end{cases} \quad (42)$$

。我们可以在 MATLAB 中编写 α_2 ，如附录 E 所示。为了说明用于 SSWMR 的 LM 训练的 DNN 控制和 RAC 策略的合成，展示了算法 4 的结构。通过对 WMR 每一侧使用建议的合成功能策略，我们可以定义 (27) 中提供的相同二次函数，从切换时间 t_s 开始分析系统稳定性，如下所示，

由 (7) 和 (41)，我们有由 (??)，并且在切换时间 t_s 时，我们有基于假设 II.3，我们可以假设存在一个正参数 f_i^* ，使得基于备注 IV.1、(??) 和 (??)，并且知道 $\bar{\theta}_i = \hat{\theta}_i - \theta_i^*$ ，我们有让我们定义一个未知的正 θ_i^* 为因此，从 (??) 和 (??)，我们有基于引理 IV.1，我们有显然，类似于 (38)，我们得到最后，我们有其中 $\mu_i = \min[A_i k_i, \delta_i]$ 和 $\ell_i = \frac{1}{4\epsilon_i} + \frac{1}{2} A_i \delta_i \theta_i^{*2}$ 。因此，基于所得的方程 (??) 和 [12], [17], [42]–[45]，所提议的用于 SSWMR 系统的 DNN 和 RAC 策略综合，如方程 (7) 所述，即使考虑切换时间，也是统一指数稳定的。

图 4 展示了所研究的 SSWMR 提出的安全控制架构的层次结构。该框架整合了两个安全层——低级安全层和高级安全层——以确保在正常和受扰动情况下稳定运行。在正常情况下，经过 LM 训练的 DNN 控制策略作为主要控制器，基于参考跟踪误差提供准确的控制命令。其输出由锁定的安全层持续监控，该层利用实时传感器反馈检测性能偏差。如果预定义的控制指标被违反，则触发低级安全层。如果干扰是可补偿的，则位于高级安全层的 RAC 策略被激活以接管控制并增强鲁棒性。然而，如果干扰超过 RAC 的补偿能力，则基于持续的安全违规，触发高级安全层关闭以防止系统或环境损害。这将导致安全终止，如图中红色所示，在正常和不利的操作场景中维护关键安全保证。

III. 实验有效性

在本节中，我们通过在在一个 6,000 公斤重的 PMSM 驱动重型 SSWMR（称为多用途布放器，MPD）上的实现来验证所提出的控制策略。机器人的每一侧都具有安装在摆架上的轮子，这些轮子在移动过程中保持与地面的接触。MPD 的机械结构和传感器设置与图 ?? 和 1 中所示相同。电气系统包括一个由电池供电的逆变器，可以精确调节每个 PMSM 的速度和转矩。液压系统则将动力从泵传递到液压马达。每个液压马达都配有里程传感器，并通过平均每侧两个轮子的速度来确定该侧在每个时间步的测量速

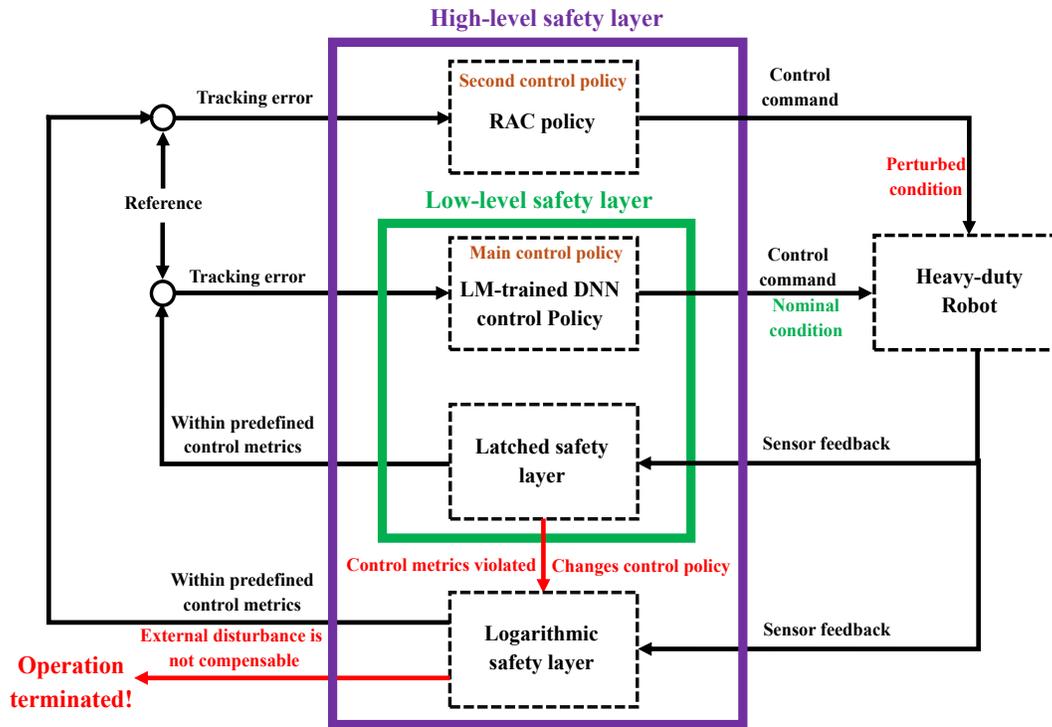


Fig. 4: 在名义和非名义条件下，由两个具有不同权限的安全层监控的 SSWMR 的 DNN 和 RAC 策略合成示意图。



Fig. 5: 6000 公斤 SSWMR 在一个安全层下使用 LM 训练的 DNN 控制策略（实验 1）和安全 RAC 策略（实验 2）跟踪的圆形轨迹（6 米半径）。

度。此过程涵盖了能量转换、高扭矩电机操作、高比例齿轮传输，最终实现车轮驱动。一个嵌入式的 Beckhoff PC 通过 EtherCAT 协议处理传感器数据采集。在控制过程中，我们假设对 MPD 的所有建模细节都是未知的。系统仅能访问来自每一侧的速度测量值，并可以通过它的 rpm 输入来控制 PMSM。

对 MPD 的每一侧，基于算法 1，我们首先通过略微增加控制输入 n_{pR} 和 n_{pL} （PMSM 转速）并在各种操作场

景中记录安装传感器测得的相应实际线速度 v_R 和 v_L ，收集了一个包含 1,000,000 对同步输入输出的数据集。这些记录的数据作为模型训练的基础。在训练模型的输入端，我们使用安装在轮毂液压电机上的 Danfoss EMD 速度传感器记录每条履带的线速度。对于输出端，我们记录每侧 PMSM 驱动泵执行器的控制信号，即泵轴转速（rpm）。所有测量值都以 0.001 秒（1 kHz）的固定采样间隔进行了时间戳，由 Beckhoff IPC CX2030 实时处理。然后，将原始

Algorithm 4 : Safe synthesis control policy for each side of the SSWM

Input : PPC metrics for both safety layers o and ζ , reference velocities v_{ref_i} , sensor feedback v_i , design parameters δ_i, γ_i, k_i
Output : Control command $u_{c_i} = n_{p_i} = \alpha_1 u_{DNN_i} + \alpha_2 u_{s_i}$

```

1  for each side  $i$  do
2    Initialize gating flags:  $\alpha_1 \leftarrow 1, \alpha_2 \leftarrow 0$  (latch-on logic)
3    Compute tracking error:  $e_i = v_i - v_{\text{ref}_i}$ 
4    Compute low-level safety:  $\zeta = (\zeta^{\text{shoot}} - \zeta^{\text{bound}}) e^{-\zeta^* t} + \zeta^{\text{bound}}$ 
5    Compute  $R = \zeta^2 - e_i^2$ 
6    if  $R \leq 0$  then (latch-off logic)
7       $\alpha_1 \leftarrow 0, \alpha_2 \leftarrow 1$  (switch to RAC policy)
8      Compute tracking error:  $e_i = v_i - v_{\text{ref}_i}$ 
9      Compute high-level safety:  $o = (o^{\text{shoot}} - o^{\text{bound}}) e^{-o^* t} + o^{\text{bound}}$ 
10     Compute denominator:  $\text{denom} = o^2 - e_i^2$ 
11     if  $\text{denom} \leq 0$  then
12       Raise safety violation and go to step 26
13     else
14       Compute safety metric:  $\log\left(\frac{o^2}{o^2 - e_i^2}\right)$ 
15     Update adaptive parameter:  $\hat{\theta}_i = -\delta_i \hat{\theta}_i + \gamma_i \left(\frac{e_i}{o^2 - e_i^2}\right)$ 
16     Compute RAC control:  $u_{s_i} = -\frac{1}{2} k_i e_i - \gamma_i \frac{e_i}{o^2 - e_i^2} \hat{\theta}_i$ 
17     Set control command:  $u_{c_i} = u_{s_i} = n_{p_i}$ 
18     Return to step 8
19   end if
20   else if
21     Keep  $\alpha_1 = 1$  and  $\alpha_2 = 0$ 
22     Pass  $v_{\text{ref}_i}$  to LM-trained DNN policy to compute  $u_{DNN_i}$ 
23     Set control command:  $u_{c_i} = u_{DNN_i} = n_{p_i}$ 
24     Return to step 4
25   end if
26   terminate operation
27 end for

```

- ζ corresponds to the low-level safety layer, while o represents the high-level saf

- Once $R \leq 0$ is detected, the system permanently switches to RAC for safety (l

- If RAC is active, the safety metric ensures operations remain within safe bound

速度和转速流对齐，清除明显的异常尖峰，并在归一化之前分为训练、验证和测试子集。

其次，从原始 $1 \times 1,000,000$ 数据向量 v_i （传感器读取值）和 n_{p_i} （控制信号）开始，我们首先使用 MATLAB 的 `dividerand` 函数随机将样本索引分为 70% 训练集、15% 验证集和 15% 测试集（见附录 A）。然后，提取了相应的输入/输出训练、输入/输出验证和输入/输出测试子集，仍然保持它们的原始（“原始”）单位。接下来，我们仅将输入-输出训练集标准化到 $[-1, +1]$ 范围，并使用 MATLAB 的 `mapminmax` 函数保存缩放参数。我们应用这些相同的参数将验证和测试输入映射到相同的标准化空间。

第三，我们在 MATLAB 中构建了 LM 训练的 DNNs，具有五个大小为 $[30, 25, 15, 10, 5]$ 的隐藏层，使用隐藏层中的默认 `tan-sigmoid` 激活函数和线性输出神经元。所有内建的输入/输出处理函数都被禁用，因为我们直接向网络输入了预先缩放的数据。训练采用了缩放共轭梯度 (`trainscg`) 算法，它在不存储完整 Hessian 的情况下提供了快速的类似二阶的收敛。我们将均方误差的性能目标设为 1×10^{-3} ，最小梯度阈值为 1×10^{-4} ，并将训练迭代的最大次数限制为 200 次。通过 70/15/15 的训练/验证/测试分割启

用了验证检查；因此，网络还监控验证 MSE 以中途停止（最多允许默认的 6 次失败），以防止过拟合。

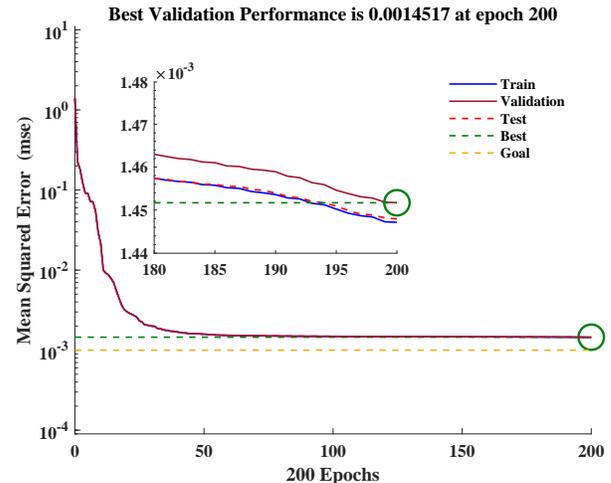


Fig. 6: LM 训练 DNN 在右侧数据上的表现。

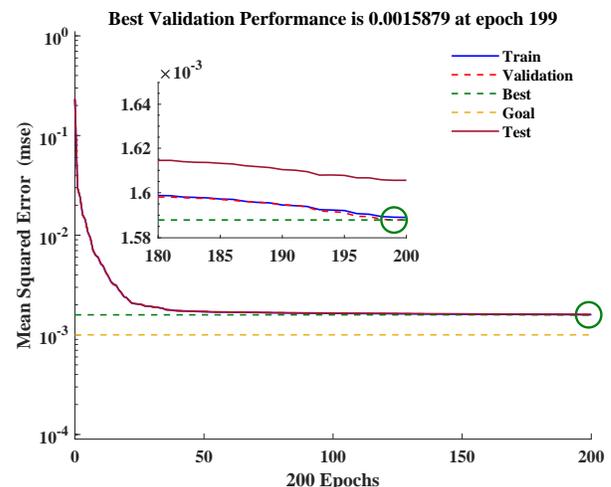


Fig. 7: LM 训练 DNN 在左侧数据上的表现。

图 6 - 13 显示了在模拟环境中两次训练过程后的相关图。图 6 和 7 表明在超过 200 个周期内，网络的训练、验证和测试均方误差在最初的几十次迭代中迅速下降，然后逐渐趋于 1.5×10^{-3} 左右，其中最佳验证误差 (≈ 0.00145) 在第 200 个周期发生；验证和测试曲线紧跟训练曲线，没有出现向上转折，表明稳定学习且没有过拟合，由于两个模型尚未明显停滞或开始偏离，因此既训练充分又稳定。

图 8 和 9 分别展示了 WMR 训练过程中右侧和左侧的误差直方图。这两个分布都紧密集中并且近似对称于零，这表明模型表现平衡且偏差很小。在图 8 中，大多数目标-输出残差落在 ± 0.02 范围内，峰值出现在以零为中心的箱中，并与蓝色无误差线对齐。类似地，图 9 显示了一个峰值非常尖锐的分布，误差范围更窄，大部分在 ± 0.01 内，且峰值稍微偏离零但仍然紧密靠近无误差参考线。

在这两种情况下，训练（白色）、验证（黑色）和测试（红色）误差在各个箱中一致地堆叠，表明模型在所有数据分割中均表现出良好的泛化能力且方差较低。虽然右侧模

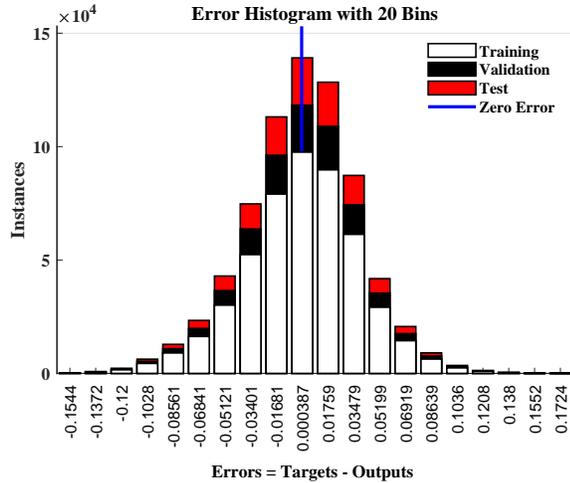


Fig. 8: LM 训练 DNN 以处理右侧数据的历史。

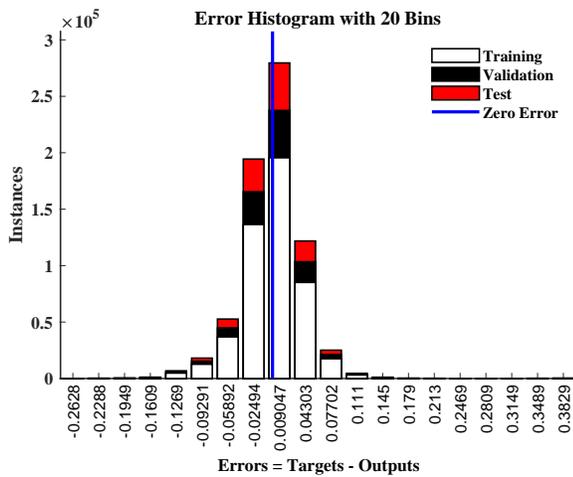


Fig. 9: 左侧数据的 LM 训练深度神经网络的历史。

型的误差很少超过 ± 0.05 ，但左侧模型显示出更少的异常值，几乎没有超出 ± 0.1 的。总体而言，两个模型都表现出强大的准确性、较低的误差幅度和一致的性能，在数据分区中，左侧模型呈现出稍微集中的误差分布。

图 10 和 11 分别显示了 WMR 训练中右侧和左侧的回归图。在这两个图中，预测输出与理想的 45° 参考线（虚线）高度一致，所有子集——训练、验证和测试——都表现出预测值与目标值之间强烈的线性关系。在两种情况下，相关系数都非常高，右侧为 $R \approx 0.9945$ ，左侧稍高为 $R \approx 0.9955$ ，表明各模型具有同样强的预测性能。两幅图中的拟合线斜率都接近 0.99，截距接近零，表明低估偏差极小。嵌入图进一步突出了残差紧密聚集在理想线附近。值得注意的是，左侧模型（图 11）实现了稍微更好的拟合，这从其略高的 R 值和更紧密集中的残差可以看出。然而，两个模型都表现出在训练、验证和测试集上的出色准确性和泛化能力，没有过拟合或系统性偏差的迹象。

图 12 和 13 分别展示了 WMR 模型的右侧和左侧在 200 个周期内的训练动态。在两种情况下，上部图显示出梯度幅度的稳定且显著的下降，最后收敛到训练结束时的 10^{-3} 量级。这表明优化是有效的，且具有稳定的梯度下降行为。

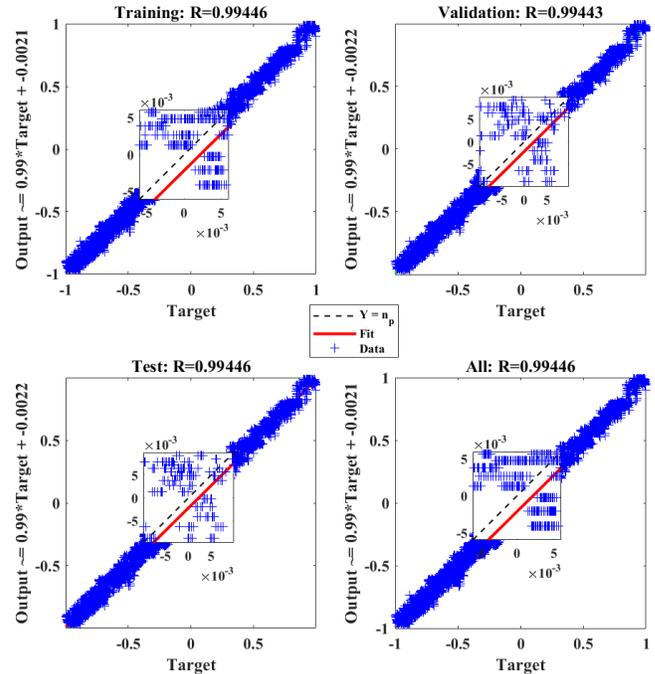


Fig. 10: 右侧数据的 LM 训练 DNN 回归。

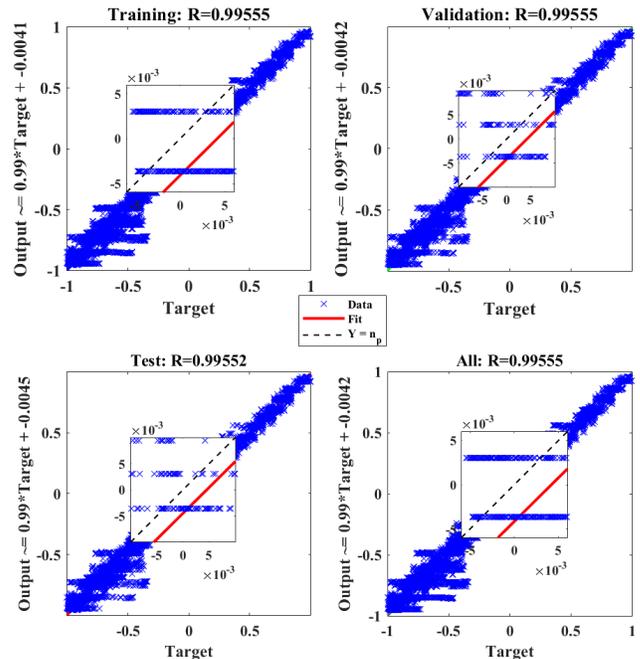


Fig. 11: 左侧数据的 LM 训练 DNN 回归。

具体来说，右侧模型的最终梯度大约为 0.00091，左侧模型为 0.00124，这两个值都足够小，表明已达到收敛。下部子图显示了整个周期中的验证检查指示器。对于右侧模型（图 12），训练期间没有发生验证失败，表明该模型在验证集上维持了一致的泛化能力，没有出现任何回归。然而，左侧模型（图 13）在接近第 100 个周期时出现了单次验证失败，但这一独立事件并未导致早停，因为模型仍

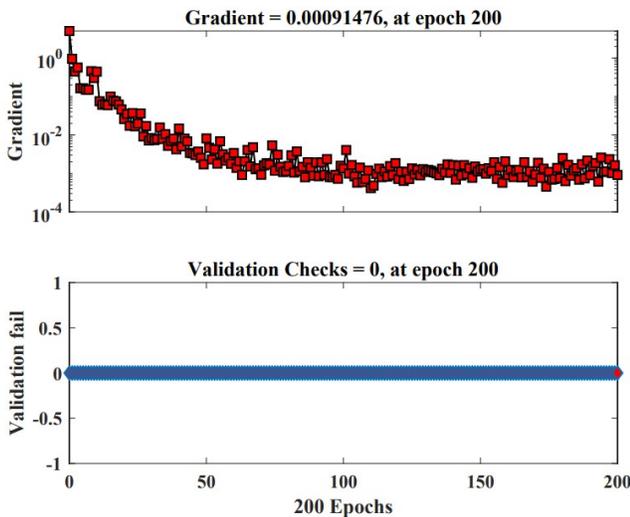


Fig. 12: 训练语言模型的状态，右侧数据的深度神经网络培训。

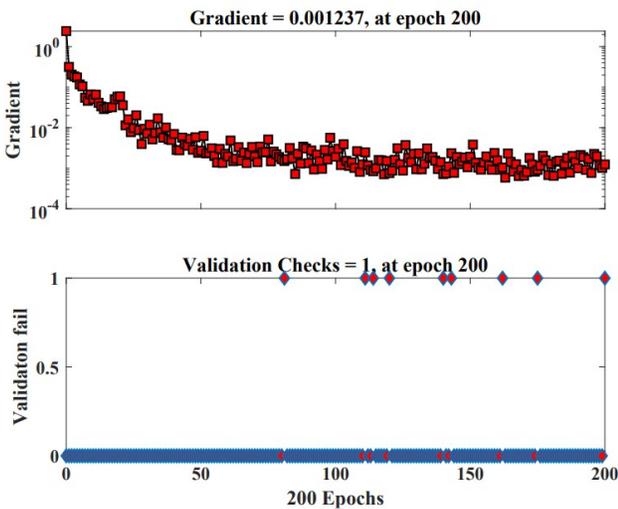


Fig. 13: 训练状态的 LM 训练 DNN 用于左侧数据。

继续训练到 200 个周期。两个模型中的验证检查次数极少（零或一），支持了平滑稳定训练且无显著过拟合的观察。

为了评估实际性能，部分 ?? 中为 MPD 的右侧和左侧训练好的基于 DNN 的控制策略被部署到一个近似半径为 6 米的圆形路径上（见图 5）。每侧的速度参考是使用文献 [11] 中描述的方法生成的。图 14 显示了在训练好的基于 DNN 的控制政策下工作的 WMR 的要求轨迹。机器人从原点 (0,0) 开始，沿着一条平滑、连续的路径运行，完成直径约为 12 米的循环。虚线黑色线条指示了机器人应行走的路径。两个侧向特定的训练好的 DNN 控制策略直接接收到命令的轮子速度，并负责生成跟踪这些速度所需的相应的 PMSM 控制输入。

为了确保安全操作（见算法 2），采用了安全层 ζ 来保证系统在预定义的安全区域内运行。随后，图 15 展示了使用双侧训练的 DNN 策略下，MPD 的跟踪误差，并在定义的性能边界下分析。左侧和右侧的跟踪误差， e_L （黄色）和 e_R （栗色），随着时间 200 秒绘制出来。图中包括时间变化的性能边界：下限 $\zeta = -0.02e^{-0.1t} - 0.02$ 和

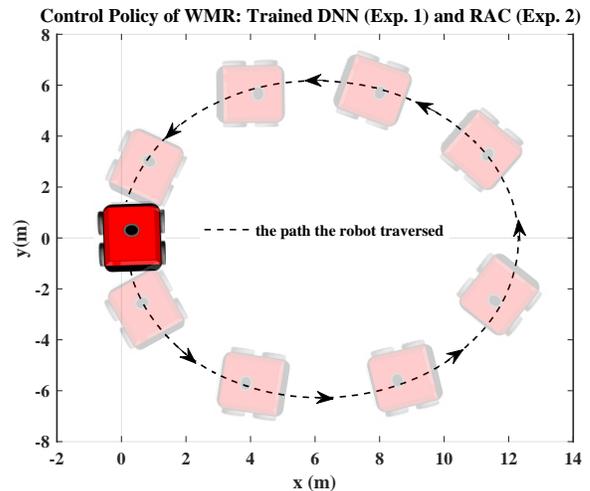


Fig. 14: 实验 1 和实验 2 中的操作任务。

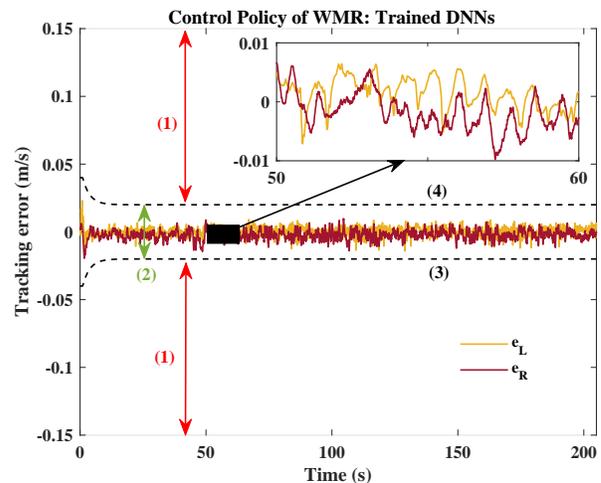


Fig. 15: 实验 1. 基于 LM 训练的 DNN 控制策略的跟踪误差。(1) : 禁止性能, (2) : 允许性能, (3): 控制策略 $\zeta = -0.02e^{-0.35t} - 0.02$ 的允许限度, (4): 控制策略 $\zeta = 0.02e^{-0.35t} + 0.02$ 的允许限度。

上限 $\zeta = 0.02e^{-0.1t} + 0.02$ ，以黑色虚线曲线（标签 3 和 4）表示。这些边界之间的中心区域表示允许的性能，而外部区域被标记为禁止区域。在图 14 所示的圆形操作期间，误差保持在安全区域内，没有任何违例，这验证了训练模型的显著准确性。插图在 50 到 60 秒之间的放大显示了精细的行为，证明了这种复杂且沉重的 WMR 的完美跟踪控制。

图 16 和 17 展示了在训练过的 DNN 下，轮式移动机器人的左右侧速度跟踪性能。在这两种情况下，把实际车轮速度 (v_L 和 v_R (实线蓝色) 与其各自的参考信号 (v_{RefL} 和 v_{RefR} , 虚线红色) 在 200 秒的期间进行了对比。尽管控制策略是黑箱的形式，两个侧面的控制器都达到了优秀的跟踪性能。在最初的瞬态阶段后，速度稳定下来并在整个实验中紧密跟随其参考信号。每个图中的小插图显示了最初几秒的情况，揭示了高保真的跟踪，偏差极小，尤其是在左侧控制器中，表现出稍微更平滑的收敛。右侧控制器也保持了较强的对准，尽管由于任务中曲线方向的特性，

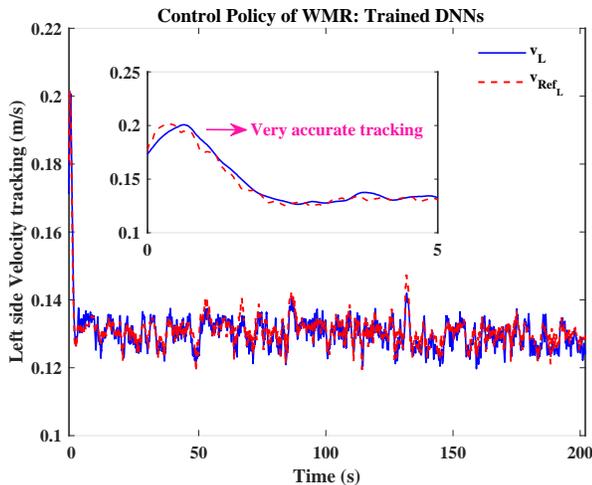


Fig. 16: 实验 1: 在左侧进行 LM 训练的 DNN 控制策略下的跟踪

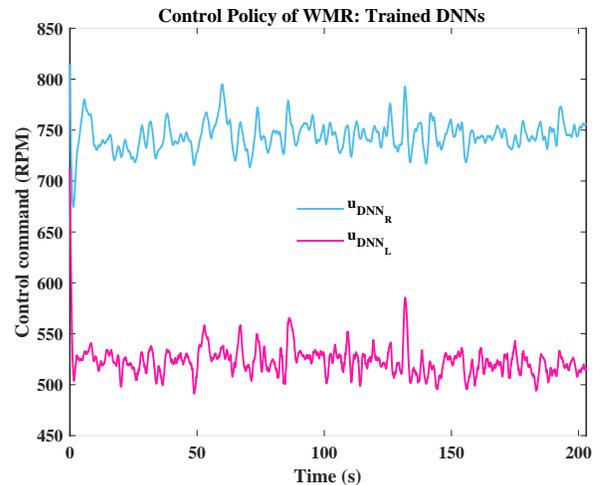


Fig. 18: 实验 1. LM 训练的 DNN 控制策略下的控制命令 (PMSM 输出)。

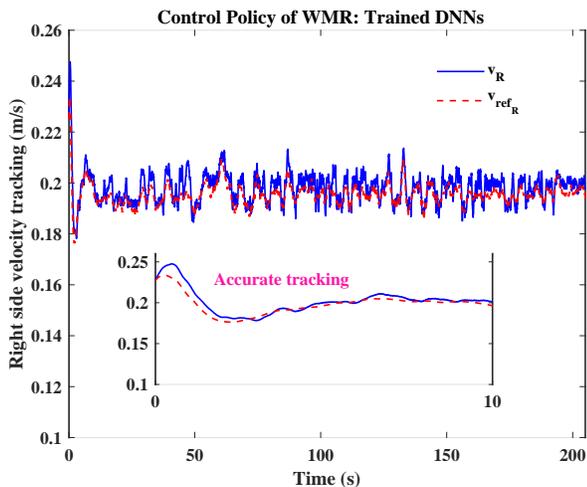


Fig. 17: 实验 1. 在 LM 训练的 DNN 控制策略下, 在右侧进行跟踪

呈现出稍微更多的振荡行为。总体而言, 这些结果展示了基于 DNN 的控制策略在一般条件下 (没有严重干扰或故障的情况下) 有效推广到轮式移动机器人的两侧, 实现了准确和稳定的速度跟踪的能力。

图 18 展示了由 LM 训练的 DNN 控制策略为 WMR 的左右轮生成的控制输入信号 (PMSM 输出)。该图显示了在 200 秒的时间跨度内 u_{DNN_R} (右轮, 淡蓝色) 和 u_{DNN_L} (左轮, 品红色) 的转速。两个信号都表现出稳定、有界的行为, 围绕各自均值约 730 rpm (右轮) 和 520 rpm (左轮) 有适度的波动 (右侧的更大努力帮助机器人沿左向轨迹行驶)。这表明需要的动态特性, 同时避免了突然或不稳定的变化。

A. 实验 2: 圆形路径的安全 RAC 政策

类似于第 ?? 节, 为了评估实际性能, 所提出的安全 RAC 方案也部署在实验 WMR 平台上, 以便遵循 Figs. 5 和 14 所示的相同圆形路径。图 Fig. 19 中的图表展示了以栗色和黄色分别表示的左右车轮跟踪误差, e_L 和 e_R 。

图表叠加了性能界限: 内虚线代表了由指数衰减函数 $\sigma = -0.06e^{-0.1t} - 0.04$ 和 $\sigma = 0.06e^{-0.1t} + 0.04$ 定义的时间变化允许极限, 随着时间的推移逐渐收紧。

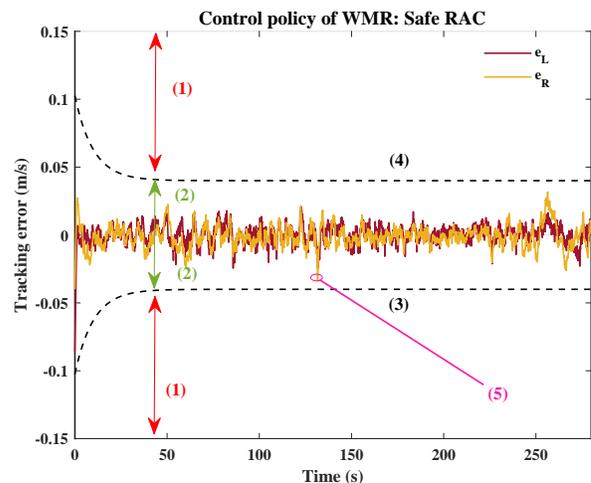


Fig. 19: 实验 2. 安全 RAC 策略下的跟踪误差。(1): 禁止性能, (2): 允许性能, (3): 控制策略 $\sigma = -0.06e^{-0.1t} - 0.04$ 的允许限, (4): 控制策略 $\sigma = 0.06e^{-0.1t} + 0.06$ 的允许限, (5): 外部干扰在 130 秒时出现时跟踪误差响应的行为。

这些界限之外的区域被标记为禁用性能区, 而界限之间的区域被标记为可允许的。在 $t = 130$ s 时, 通过将控制输入相对于指令值增加 8%, 引入了两次小的外部扰动。尽管有这些扰动, 跟踪误差曲线在整个实验中都保持在定义的性能界限之内。这表明安全的 RAC 策略在补偿扰动的同时, 遵循了性能约束并将跟踪误差保持在安全范围内。如前所述, 尽管 RAC 方案中的自适应律可以提高对不确定性和外部扰动的鲁棒性, 但代价是相对于图 15 所示的基于 DNN 的策略而言控制精度降低, 这凸显了鲁棒性与响应性之间的权衡。这在图 19 中很明显, 图中显示安全 RAC 方案下的跟踪误差高于经过训练的 DNN 策略。因此, RAC 的性能界限被设定得比用于 DNN 方法的宽。图

20 和 21 中得出的结论类似。这些图展示了 WMR 平台在所提出的安全 RAC 方案下左、右轮速度跟踪性能。

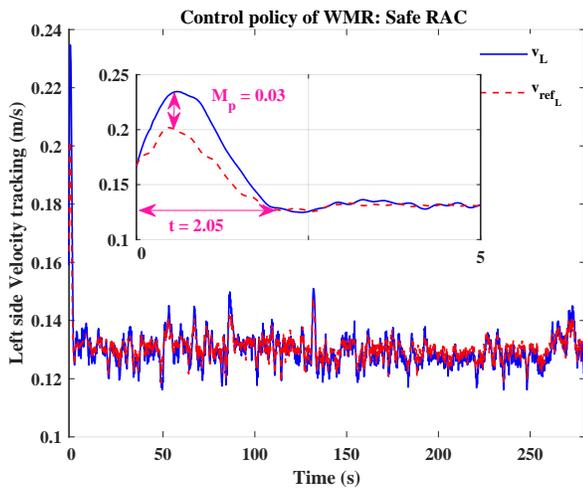


Fig. 20: 实验 2. 在安全 RAC 策略下跟踪左侧

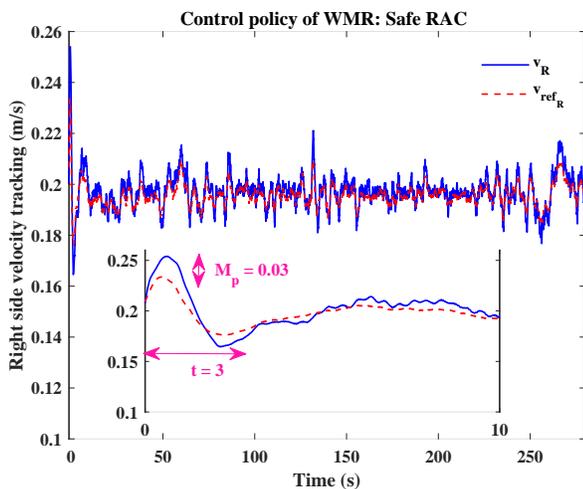


Fig. 21: 实验 2. 在安全 RAC 政策下左侧的跟踪

这两幅图表现出相似的趋势。两幅图都表现出一个初始瞬态阶段，特征是轻微超调和较短的稳定时间。具体来说，图 20 显示左轮有大约为 0.03 m/s 的超调 (M_p) 和大约 2.05 秒的稳定时间，而图 21 显示右轮具有相同超调幅度，但稳定时间稍长，为大约 3 秒。在稳态阶段（瞬态之后），实际速度紧密跟踪其参考值，只有轻微的震荡和波动，可能是由于自适应控制律来补偿未建模动态。这两幅图还揭示，尽管跟踪性能是可接受的，但是控制特性——尤其是超调和瞬态响应——比经过训练的基于 DNN 的控制策略略弱，正如图 16 和 17 所示。图 22 描绘了安全 RAC 控制器为 WMR 的左轮和右轮生成的控制命令。尽管存在一些噪声和波动，控制信号总体上是平滑和稳定的，表明安全 RAC 策略的有效调节。

正如在第 ?? 节和第 III-A 节中所示，训练的 DNN 控制策略相比于稳定性保证的 RAC 展示了更好的控制精度。此外，正如第 ?? 节中提到的，实际上具有复杂驱动机制的重型 WMR 主要在标称条件下运行，其中提出的 LM 训练的 DNN 控制策略在第 ?? 节中展示了高性能。该控

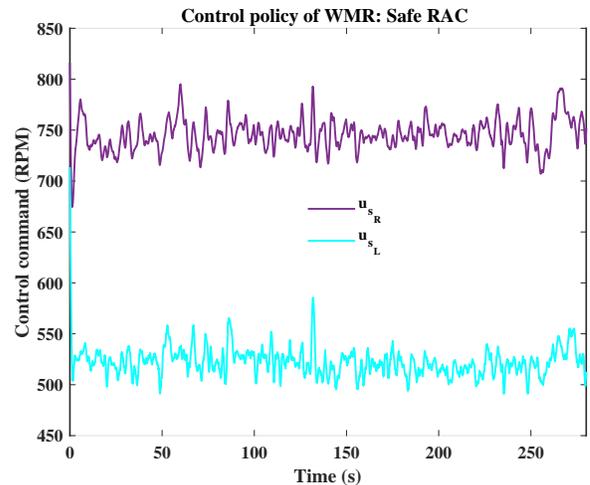


Fig. 22: 实验 2. 在安全的 RAC 策略下的控制命令 (PMSM 输出)。

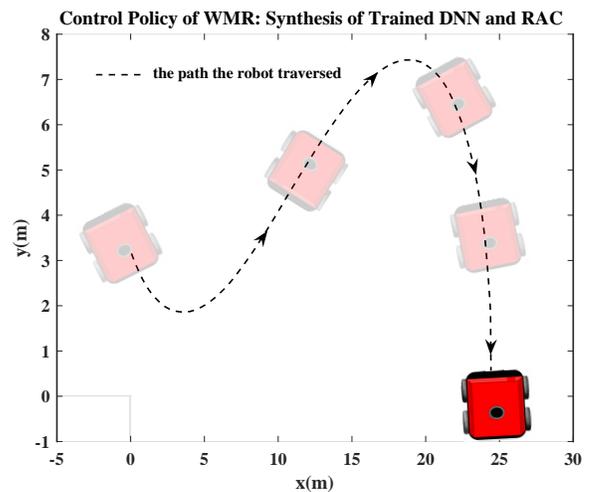


Fig. 23: 实验 3 中的操作任务。

制策略可以容忍轻微或瞬态的外部干扰而不会显著降低性能。然而，为了确保安全可靠地运作——特别是在涉及严重干扰或故障的罕见但关键的场景中——必须保证符合严格国际安全标准的鲁棒性和系统稳定性。因此，在研究的 MPD 中结合了这两种控制方法，以确保不仅有精度而且还有鲁棒性（见算法 4）。通过集成稳定性保证的 RAC 策略，系统受益于两个独立的安全层，即使训练好的 DNN 控制策略未能对潜在的干扰攻击作出反应，仍能保持持续运行。该方法允许在系统全面检修的适当时间之前保持功能持续。为了评估实际性能，第 ?? 节中提出的控制方法在实验 WMR 平台上部署，以跟随图 23 所示的 S 形路径。每个车轮的速度参考根据参考文献 [11] 中描述的方法生成。图 25 显示了在 LM 训练的 DNN 和安全 RAC 策略的组合控制下 WMR 的跟踪误差。该图展示了 180 秒期间左轮和右轮， e_L 和 e_R 的跟踪误差。最初，MPD 在 DNN 控制策略下运行，通过性能界限一由虚线蓝线表示，定义为 $\zeta = -0.02e^{-0.35t} - 0.02$ 和 $\zeta = 0.02e^{-0.35t} + 0.02$ ——提供强大的性能。在 $t = 82$ 秒时，我们通过将控制输入增加 15% 而不是要求值，引入了两个侧面的中等信号作



Fig. 24: 实验 3: S 形轨迹由 SSWMR 跟踪, 该过程使用 LM 训练的 DNN 的综合和在两个安全层下的安全 RAC 策略。

为外部干扰到操作系统中。

扰, 验证了监管架构的有效性。

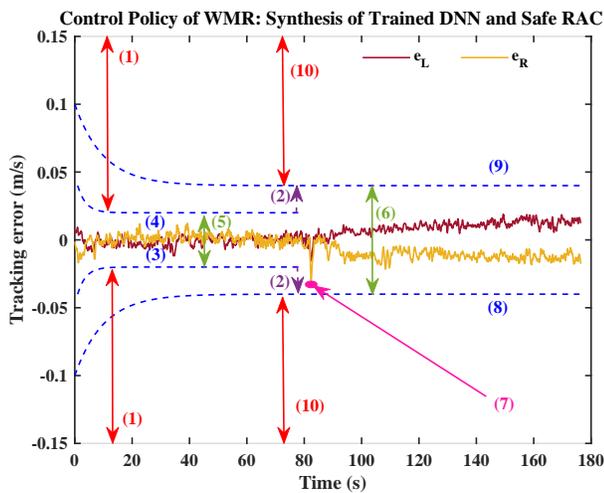


Fig. 25: 实验 3. 在 LM 训练的 DNN 和安全 RAC 政策综合下的跟踪误差。(1) : DNN 控制策略的禁止性能, (2) : 由于外部干扰而激活的切换控制 (超出低级安全层), (3) : DNN 策略 $\zeta = -0.02e^{-0.35t} - 0.02$ 的允许下限, (4) : DNN 策略 $\zeta = 0.02e^{-0.35t} + 0.02$ 的允许上限, (5) : 低级安全层的允许性能, (6) : 高级安全层的允许性能, (7) : 当外部干扰在 82 秒处发生时, 跟踪误差响应的行为, (8) : RAC 政策 $o = -0.06e^{-0.03t} - 0.04$ 的允许下限, (9) : RAC 政策 $o = 0.06e^{-0.03t} + 0.04$ 的允许上限, (10) : RAC 控制策略的禁止性能。

这导致利用 LM 训练的 DNN 控制跟踪误差超出了底层安全层的界限。因此, 系统在大约 80.001 秒时转向稳定性有保证的 RAC 策略, 保持在高层安全层下的运行。RAC 策略施加了它自己的界限: $o_i = -0.06e^{-0.04t} - 0.03$ (7) 和 $o_i = 0.06e^{-0.03t} + 0.04$ (8), 这些界限更为宽松以应对干扰。该图展示了控制器能够切换模式以保持安全和适应干

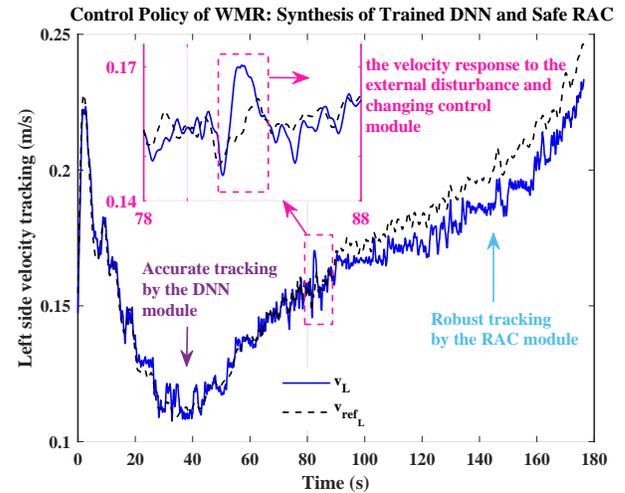


Fig. 26: 实验 3. 在 LM 训练的 DNN 和安全 RAC 策略的综合下进行左侧跟踪。

图 26 显示了本次实验中左侧的速度跟踪性能。最初, DNN 策略提供了准确的跟踪, 误差极小, 如注释所示。在大约 82 秒时, 出现了外部干扰, 导致了瞬态偏差。这触发了从 DNN 到 RAC 策略的锁定功能切换, 由方框标出的区域和 78 到 88 秒之间的速度波动 (速度在 0.14 到 0.17 m/s 之间变化) 所突出显示。过渡后, RAC 策略接管并提供稳健的跟踪, 适应干扰的同时保持与参考的对齐。

WMR 控制系统中结合了基于 DNN 的控制策略和安全 RAC 策略的左右两侧的速度跟踪性能在时间上表现出一致的结构: 在标称阶段 (~ 82 s 之前), DNN 策略进行精确的跟踪; 在由 RAC 策略管理的鲁棒恢复阶段 (~ 82 s 到 84 s); 以及在 RAC 策略控制下的持续跟踪 (84 s 之后), 虽然准确性有所降低。右侧通常表现出更高的速度幅度、恢复速度更快以及扰动后的平滑过渡, 而左侧则显示

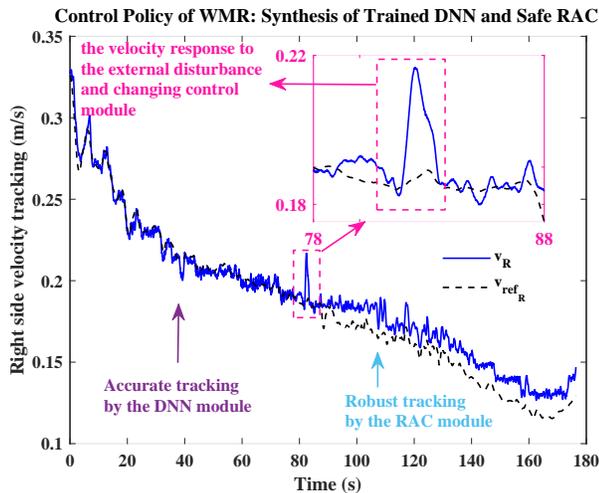


Fig. 27: 实验 3. 在 LM 训练的 DNN 与安全 RAC 策略合成下的右侧跟踪。

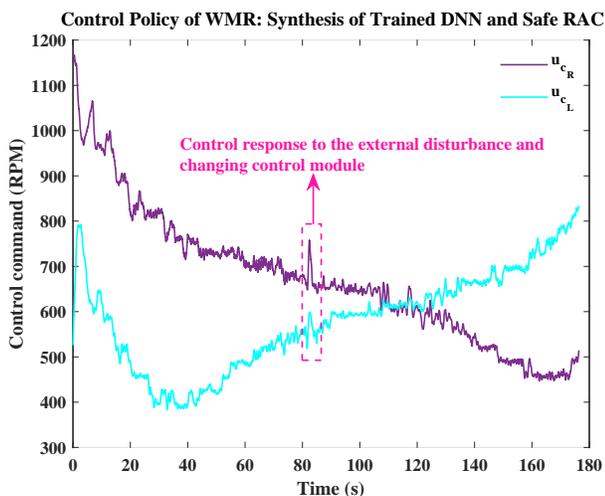


Fig. 28: 实验 3. 在 LM 训练的 DNN 和安全 RAC 策略合成下的控制命令 (PMSM 输出)。

出稍多的振荡和较慢的稳定。总体而言，这两个图验证了在具有不同控制权的两个安全层中所提出的控制策略能够有效管理系统响应性和鲁棒性之间的权衡。

图 28 展示了在提出的控制架构下，WMR 左右两侧 (PMSM 输出) 的控制命令。信号 u_{c_L} (淡蓝色) 和 u_{c_R} (深黑色) 代表在 180 秒内电机的转速输入。最初，两个控制信号与 DNN 控制策略 (u_{DNN_i}) 相对应，在标称条件下管理系统直到 $t = 82$ 秒。在此之后，控制切换到 RAC 策略 (u_{s_i})，该策略补偿了受扰动的运行。这一过渡由控制输入中的一个小尖峰标记，表明对干扰的纠正措施。

本文提出了一种实用且理论上有所依据的解决方案，用于在复杂和不可预测的条件下安全稳健地控制重型轮式移动机器人，该方案受到 ISO/IEC TR 5469 —— 人工智能——功能安全与 AI 系统的启发。为了管理系统稳健性与响应性之间的基本权衡，设计了一种分层控制策略，然后通过两个权威性不同的安全层进行监控。在正常条件下，LM 训练的 DNN 控制策略作为主要控制器，提供高精度的性能。当外部干扰发生并且变得足够强烈以致控制性能不再

符合预定义的低级安全层时，条件锁存功能将停用 LM 训练的 DNN 控制策略并激活稳定性保证的 RAC 策略作为替代，从而确保持续安全的操作。在这两种控制策略期间，高级安全层持续监控系统性能，只有当干扰严重到无补偿可能且继续运行会对系统或其环境构成风险时，才触发停机。所提出的架构确保了 WMR 整个框架的统一指数稳定性。通过对单独的基于 DNN 的策略、独立的 RAC 策略及其在配备五级执行机制的 6,000 公斤 WMR 上的集成进行逐步实验评估，进一步验证了控制框架。比较结果证明了所有提出的方法在现实场景中的可行性。这种方法有可能为在重型机器人系统中更广泛使用学习算法铺平道路，同时仍符合严格的国际标准。

为了在 MATLAB 中进行数据准备，我们定义了输入和目标信号，并使用 dividerand 将数据分成训练、验证和测试集，如

```

1 X = v_i;
2 T = n_{p_i};
3 N = size(X, 2);
4 [trainInd, valInd, testInd] = dividerand(N, 0.70,
5     0.15, 0.15);
6 Xtrain = X(:, trainInd); Ttrain = T(:, trainInd);
7 Xval = X(:, valInd); Tval = T(:, valInd);
8 Xtest = X(:, testInd); Ttest = T(:, testInd);

```

Listing 1: Raw data preparation and splitting

然后，输入和目标使用 mapminmax 进行归一化，以提高训练性能。参数仅从训练集计算，以避免数据泄露。

```

1 [XtrainN, inputPS] = mapminmax(Xtrain, -1, 1);
2 XvalN = mapminmax('apply', Xval, inputPS);
3 XtestN = mapminmax('apply', Xtest, inputPS);
4 [TtrainN, outputPS] = mapminmax(Ttrain, -1, 1);
5 TvalN = mapminmax('apply', Tval, outputPS);
6 TtestN = mapminmax('apply', Ttest, outputPS);

```

Listing 2: Input and target normalization using mapminmax

作为网络定义和数据分配的一部分，创建了一个具有五个隐藏层的前馈神经网络。内置预处理被禁用，并手动为每个子集指定数据索引。

```

1 hiddenSizes = [30 25 15 10 5];
2 net = feedforwardnet(hiddenSizes, 'trainscg');
3 net.inputs{1}.processFcns = {};
4 net.outputs{end}.processFcns = {};
5 net.divideFcn = 'divideind';
6 net.divideParam.trainInd = 1 : numel(trainInd);
7 net.divideParam.valInd = numel(trainInd) + (1:
8     numel(valInd));
9 net.divideParam.testInd = numel(trainInd) + numel(
10    valInd) + (1:numel(testInd));
11 XallN = [XtrainN, XvalN, XtestN];
12 TallN = [TtrainN, TvalN, TtestN];

```

Listing 3: Network creation and configuration

训练参数如目标、最小梯度和最大训练次数被设置好。然后在完整的标准化数据集上训练网络。

```

1 net.trainParam.goal = 1e-3;
2 net.trainParam.min_grad = 1e-4;
3 net.trainParam.epochs = 200;
4 [net, tr] = train(net, XallN, TallN);

```

Listing 4: Training parameters and network training

最后，训练好的网络将在所有数据子集上进行评估。预测结果被反归一化并与真实值进行比较。最终的 MSE 值将被打印。

```

1 trainIdx = net.divideParam.trainInd;
2 valIdx   = net.divideParam.valInd;
3 testIdx  = net.divideParam.testInd;
4 YtrainN = net(XallN(:, trainIdx));
5 YvalN   = net(XallN(:, valIdx));
6 YtestN  = net(XallN(:, testIdx));
7 Ytrain  = mapminmax('reverse', YtrainN, outputPS);
8 Yval    = mapminmax('reverse', YvalN, outputPS);
9 Ytest   = mapminmax('reverse', YtestN, outputPS);
10 mse_train = mean((Ytrain - Ttrain).^2);
11 mse_val   = mean((Yval   - Tval).^2);
12 mse_test  = mean((Ytest  - Ttest).^2);
13
14 fprintf('MSE (raw scale): Train =
15         mse_train, mse_val, mse_test);

```

Listing 5: Evaluate performance and compute MSE

IV.

附录 B: 基于 Blf 的神经网络策略安全层 由于对数 BLF 充当了 LM 训练的基于 DNN 的 MPD 控制策略的安全监控器, 我们可以在 MATLAB 中实现如下:

```

1 denom = (zeta)^2 - e_i.^2;
2 if any(denom <= 0)
3     error('Safety violation: e_i^2 >= (zeta)^2.
4         Operation terminated. ');
5 else
6     safety_metric = log((zeta)^2 ./ denom);
7     disp('safety_metric is defined. Operation can
8         continue. ');
9     disp(safety_metric);
10 end

```

Listing 6: Safety index using logarithmic expression for DNN controllers

V.

附录 C: 实施建议的 RAC 政策 RAC 的 MATLAB 代码如下:

```

1 tspan = [t0 tf];
2 theta0 = hat_theta_0;
3 o_func = @(t) (o_shoot - o_bound) * exp(-o_star * t)
4         + o_bound;
5 dtheta_dt = @(t, theta) -delta_i * theta + ...
6             gamma_i * (e_i / (o_func(t)^2 - e_i^2))^2;
7 [t, theta] = ode45(dtheta_dt, tspan, theta0);
8 u_si = @(t, e_i, theta) -0.5 * k_i * e_i - gamma_i *
9         (e_i / (o_func(t)^2 - e_i^2)) * theta;

```

Listing 7: A sample form of MATLAB simulation for Eqs. 1 (32) and (39)

VI.

附录 D: 基于 Blf 的 Rac 策略安全层 该对数屏障函数作为安全指标, 可以在 MATLAB 中实现如下:

```

1 denom = o^2 - e_i.^2;
2 if any(denom <= 0)
3     error('Safety violation: e_i^2 >= o^2. Operation
4         terminated. ');
5 else
6     safety_metric = log(o^2 ./ denom);
7     disp('safety_metric is defined. Operation can
8         continue. ');
9     disp(safety_metric);

```

end

Listing 8: Safety index using logarithmic expression for RAC

VII.

附录 E: 提议的使用安全 RAC 策略的 DNN 合成 如果我们定义低级安全层边际为 $R = b^2 - e_i^2$, 那么我们可以在 MATLAB 中编码 α_1 , 作为 (??)。

```

1 alpha_1 = zeros(size(R));
2 active = true;
3
4 for i = 1:length(R)
5     if active && R(i) <= 0
6         active = false;
7     end
8     alpha_1(i)(i) = double(active && R(i) > 0);
9 end

```

Listing 9: Low-level safety layer based on latch-off logic

这个代码创建了一个永久性的关机信号, 这个信号一开始为“开”(活动 = true), 并在安全条件 $R \leq 0$ 首次被违反时永久关闭。即使条件后来恢复, 系统仍然保持关闭。相反, 我们还使用 α_2 为提议的 RAC 控制器, 它在逻辑函数保持开启的起始时刻为 0, 并在条件 ($R \geq 0$) 发生时刻切换为 1, 无论之后的输入如何, 永远保持为 1。 α_2 定义为

$$\alpha_2(t) = \begin{cases} 0, & \text{if } e_i^2(\tau) < b^2(\tau) \text{ for all } \tau \in [0, t] \\ 1, & \text{if there exists } \tau \in [0, t] \text{ such } b^2(\tau) \leq e_i^2(\tau) \end{cases} \quad (43)$$

我们可以在 MATLAB 中编码 α_2 , 作为。

```

1 alpha_2 = zeros(size(R));
2 latched = false;
3
4 for i = 1:length(R)
5     if ~latched && R(i) <= 0
6         latched = true;
7     end
8     alpha_2(i) = double(latched);
9 end

```

Listing 10: Low-level safety layer based on latch-on logic

; 现在我们可以定义 u_{c_i} :

```

1 u_c_i = @(t) alpha1(t) * u_DNN_i(t) + alpha2(t) *
2         u_s_i(t);

```

Listing 11: Computation of control input u_{c_i}

REFERENCES

- [1] D. Herman, “A renaissance robot,” *Mechanical Engineering*, vol. 120, no. 02, pp. 80–82, 1998.
- [2] T. Romanenko and P. Shcherbinina, “Robot vs worker,” *Technology and language*, vol. 3, no. 1, pp. 17–28, 2022.
- [3] J. A. Adams, “Human-robot teaming directions for dull, dirty and dangerous domains,” in *2025 20th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2025, pp. 1–1.
- [4] G. Klancar, A. Zdesar, S. Blazic, and I. Skrjanc, *Wheeled mobile robotics: from fundamentals towards autonomous systems*. Butterworth-Heinemann, 2017.

- [5] T. Machado, D. Fassbender, A. Taheri, D. Eriksson, H. Gupta, A. Molaei, P. Forte, P. K. Rai, R. Ghabcheloo, S. Mäkinen *et al.*, “Autonomous heavy-duty mobile machinery: A multi-disciplinary collaborative challenge,” in *2021 IEEE International Conference on Technology and Entrepreneurship (ICTE)*. IEEE, 2021, pp. 1–8.
- [6] F. Chi, J. Zhou, Q. Zhang, Y. Wang, and P. Huang, “Avoiding the health hazard of people from construction vehicles: a strategy for controlling the vibration of a wheel loader,” *International journal of environmental research and public health*, vol. 14, no. 3, p. 275, 2017.
- [7] R. Leung, A. J. Hill, and A. Melkumyan, “Automation and artificial intelligence technology in surface mining: A brief introduction to open-pit operations in the pilbara,” *IEEE Robotics & Automation Magazine*, 2023.
- [8] Z. Li, J. Deng, R. Lu, Y. Xu, J. Bai, and C.-Y. Su, “Trajectory-tracking control of mobile robot systems incorporating neural-dynamic optimized model predictive approach,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 6, pp. 740–749, 2015.
- [9] O. Elshazly, A. Abo-Ismael, H. S. Abbas, and Z. Zyada, “Skid steering mobile robot modeling and control,” in *2014 UKACC International Conference on Control (CONTROL)*. IEEE, 2014, pp. 62–67.
- [10] A. Mazur and M. Cholewiński, “Virtual force concept in steering mobile manipulators with skid-steering platform moving in unknown environment,” *Journal of Intelligent & Robotic Systems*, vol. 77, pp. 433–443, 2015.
- [11] M. H. Shahna, E. Haaparanta, P. Mustalahti, and J. Mattila, “LiDAR-inertial SLAM-based navigation and safety-oriented AI-driven control system for skid-steer robots,” *arXiv preprint arXiv:2505.02598*, 2025.
- [12] M. H. Shahna, P. Mustalahti, and J. Mattila, “Robust torque-observed control with safe input–output constraints for hydraulic in-wheel drive systems in mobile robots,” *Control Engineering Practice*, vol. 164, p. 106459, 2025.
- [13] J. Liao, Z. Chen, and B. Yao, “Model-based coordinated control of four-wheel independently driven skid steer mobile robot with wheel–ground interaction and wheel dynamics,” *IEEE Transactions on Industrial Informatics*, vol. 15, no. 3, pp. 1742–1752, 2018.
- [14] —, “Performance-oriented coordinated adaptive robust control for four-wheel independently driven skid steer mobile robot,” *IEEE Access*, vol. 5, pp. 19048–19057, 2017.
- [15] R. Xiong, L. Li, C. Zhang, K. Ma, X. Yi, and H. Zeng, “Path tracking of a four-wheel independently driven skid steer robotic vehicle through a cascaded NTSM-PID control method,” *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–11, 2022.
- [16] E. Mohammadpour and M. Naraghi, “Robust adaptive stabilization of skid steer wheeled mobile robots considering slipping effects,” *Advanced robotics*, vol. 25, no. 1-2, pp. 205–227, 2011.
- [17] M. H. Shahna, M. Bahari, and J. Mattila, “Robustness-guaranteed observer-based control strategy with modularity for cleantech EMLA-driven heavy-duty robotic manipulator,” *IEEE Transactions on Automation Science and Engineering*, 2024.
- [18] Z. Chen, Y.-S. Hao, Z.-g. Su, and L. Sun, “Cascade disturbance observer based control design for cascaded systems with considerable inner-loop dynamics,” *IEEE Transactions on Automation Science and Engineering*, 2025.
- [19] G. Li, T. Stalin, P. V. y Alvarado *et al.*, “DNN-based predictive model for a batoid-inspired soft robot,” *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 1024–1031, 2021.
- [20] C. Badgular, S. Das, D. M. Figueroa, D. Flippo, and S. Welch, “Deep neural networks to predict autonomous ground vehicle behavior on sloping terrain field,” *Journal of Field Robotics*, vol. 40, no. 4, pp. 919–933, 2023.
- [21] D. Costello III, L. DeVries, C. Mauldin, and B. Ross, “DNN based ranging in support of autonomous aerial refueling,” *Journal of Intelligent & Robotic Systems*, vol. 109, no. 3, p. 49, 2023.
- [22] Z.-P. Kuo and J. I.-Z. Chen, “To deploy trained speech with DNN-lstm framework for controlling a smart wheeled-robot in limited learning circumstance,” *International Journal of Speech Technology*, vol. 25, no. 4, pp. 879–891, 2022.
- [23] S. Chaudhury and T. Yamasaki, “Robustness of adaptive neural network optimization under training noise,” *IEEE Access*, vol. 9, pp. 37 039–37 053, 2021.
- [24] F. Yu, Z. Qin, C. Liu, D. Wang, and X. Chen, “Rein the robots: Robust DNN-based image recognition in autonomous driving systems,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 40, no. 6, pp. 1258–1271, 2020.
- [25] G. Amir, O. Maayan, T. Zelazny, G. Katz, and M. Schapira, “Verifying the generalization of deep learning to out-of-distribution domains,” *Journal of Automated Reasoning*, vol. 68, no. 3, p. 17, 2024.
- [26] E. ŞAHİN, N. N. Arslan, and D. Özdemir, “Unlocking the black box: an in-depth review on interpretability, explainability, and reliability in deep learning,” *Neural Computing and Applications*, vol. 37, no. 2, pp. 859–965, 2025.
- [27] A. Agarwal, Y. Ruan, and L. Pileggi, “A hybrid simulation of DNN-based gray box models,” *arXiv preprint arXiv:2410.17103*, 2024.
- [28] E. ISO, “En iso 3691,” 1977.
- [29] B. W. Main and C. F. Hayes, “Control systems: Insights on iso 13849-1,” *Professional Safety*, vol. 59, no. 07, pp. 41–51, 2014.
- [30] R. Bell, “Introduction to iec 61508,” in *Acm international conference proceeding series*, vol. 162. Citeseer, 2006, pp. 3–12.
- [31] L. Wang and J. Dong, “Concurrent learning control Lyapunov and barrier functions for uncertain nonlinear safety-critical systems with high relative degree constraints,” *IEEE Transactions on Automation Science and Engineering*, vol. 21, no. 4, pp. 7170–7179, 2023.
- [32] J. Liang, H. Zhong, Y. Wang, Y. Chen, J. Zeng, and J. Mao, “Adaptive force tracking impedance control for aerial interaction in uncertain contact environment using barrier function,” *IEEE Transactions on Automation Science and Engineering*, 2023.
- [33] L. Liu, Y.-J. Liu, D. Li, S. Tong, and Z. Wang, “Barrier Lyapunov function-based adaptive fuzzy ftc for switched systems and its applications to resistance–inductance–capacitance circuit system,” *IEEE Transactions on Cybernetics*, vol. 50, no. 8, pp. 3491–3502, 2019.
- [34] “Artificial intelligence—functional safety and ai systems,” International Organization for Standardization, Tech. Rep. ISO/IEC TR 5469:2024, 2024. [Online]. Available: <https://www.iso.org/standard/81283.html>
- [35] J.-X. Zhang and G.-H. Yang, “Low-complexity tracking control of strict-feedback systems with unknown control directions,” *IEEE Transactions on Automatic Control*, vol. 64, no. 12, pp. 5175–5182, December 2019.
- [36] Y. Huang and Y. Liu, “Practical tracking via adaptive event-triggered feedback for uncertain nonlinear systems,” *IEEE Transactions on Automatic Control*, vol. 64, no. 9, pp. 3920–3927, January 2019.
- [37] S. Saha, S. Dutta, B. Goswami, and D. Nandi, “ADU-Net: an attention dense U-Net based deep supervised DNN for automated lesion segmentation of COVID-19 from chest CT images,” *Biomedical Signal Processing and Control*, vol. 85, p. 104974, 2023.
- [38] F. M. Shakiba and M. Zhou, “Novel analog implementation of a hyperbolic tangent neuron in artificial neural networks,” *IEEE Transactions on Industrial Electronics*, vol. 68, no. 11, pp. 10 856–10 867, 2020.
- [39] M. H. Shahna, P. Mustalahti, and J. Mattila, “Robust model-free control framework with safety constraints for a fully electric linear actuator system,” in *2024 IEEE 21st International Power Electronics and Motion Control Conference (PEMC)*. IEEE, 2024, pp. 1–10.
- [40] M. H. Shahna, J.-P. Humaloja, and J. Mattila, “Model reference-based control with guaranteed predefined performance for uncertain strict-feedback systems,” *arXiv preprint arXiv:2502.03263*, 2025.
- [41] K. P. Tee, S. S. Ge, and E. H. Tay, “Barrier Lyapunov functions for the control of output-constrained nonlinear systems,” *Automatica*, vol. 45, no. 4, pp. 918–927, April 2009.
- [42] M. H. Shahna and J. Mattila, “Exponential auto-tuning fault-tolerant control of n degrees-of-freedom manipulators subject to torque constraints,” in *2024 IEEE 63rd Conference on Decision and Control (CDC)*. IEEE, 2024, pp. 7263–7270.

- [43] M. Corless and G. Leitmann, “Bounded controllers for robust exponential convergence,” *Journal of Optimization Theory and Applications*, vol. 76, no. 1, pp. 1–12, 1993.
- [44] M. H. Shahna, S. A. A. Kolagar, and J. Mattila, “Integrating DeepRL with robust low-level control in robotic manipulators for non-repetitive reaching tasks,” in *2024 IEEE International Conference on Mechatronics and Automation (ICMA)*. IEEE, 2024, pp. 329–336.
- [45] H. Wu, “Robust tracking and model following of uncertain nonlinear systems with time-varying delays and dead-zone inputs,” *IET Control Theory & Applications*, vol. 14, no. 6, pp. 801–808, 2020.



ing.

Mehdi Heydari Shahna earned a B.Sc. in electrical engineering from Razi University, Kermanshah, Iran, in 2015 and an M.Sc. in control engineering at Shahid Beheshti University, Tehran, Iran, in 2018. Since December 2022, he has been pursuing his doctoral degree in automation technology and mechanical engineering at Tampere University, Tampere, Finland. His research interests encompass robust control, robotics, fault-tolerant algorithms, and system stability, Robot Learn-



Jouni Mattila received an M.Sc. and Ph.D. in automation engineering from Tampere University of Technology, Tampere, Finland, in 1995 and 2000, respectively. He is currently a professor of machine automation with the Unit of Automation Technology and Mechanical Engineering at Tampere University. His research interests include machine automation, nonlinear-model-based control of robotic manipulators, and energy-efficient control of heavy-duty mobile manipulators.