

MoRPI-PINN: 用于移动机器人纯惯性导航的物理信息框架

Arup Kumar Sahoo and Itzik Klein

Abstract—对移动机器人实现完全自主性的一个基本要求是在卫星导航或摄像头不可用的情况下也能进行精确的导航。在这种情况下，仅依赖惯性传感器会由于传感器的固有噪声和误差项而导致导航解的漂移。为了减轻漂移影响，正在兴起的一种解决方案是使机器人以蛇形爬行的方式运动，从而提高惯性信号的信噪比，以便回归移动机器人的位置。在这项工作中，我们提出了 MoRPI-PINN 作为一种物理信息神经网络框架，用于基于惯性的移动机器人精确导航。通过将物理定律和约束嵌入到训练过程中，MoRPI-PINN 能够提供精确且稳健的导航解决方案。通过实际实验，我们展示了与其他方法相比准确度提高了超过 85 %。MoRPI-PINN 是一种轻量化的方法，甚至可以在边缘设备上实施，并可用于任何典型的移动机器人应用中。

Index Terms—Scientific Machine Learning; Physics-informed Neural Networks; Inertial Navigation System; Mobile Robot; Accelerometer; Gyroscope; Yaw Angle; Dead Reckoning.

TABLE
缩略词列表

Abbreviation	Definition
Navigation	
INS	Inertial Navigation System
IMU	Inertial Measurement Unit
GNSS	Global Navigation Satellite System
RTK	Real-Time Kinematic
NED	North-East-Down (Coordinate Frame)
ECEF	Earth-Centered Earth-Fixed
Neural Networks	
PINN	Physics-informed Neural Network
AD	Automatic Differentiation
GT	Ground Truth
Error Metrics	
ATE	Absolute Trajectory Error
MATE	Mean Absolute Trajectory Error
MSE	Mean Squared Error
RMSE	Root Mean Squared Error
NRMSE	Normalized Root Mean Squared Error

I. 介绍

他 移动机器人的研究对于现代工业和服务至关重要，因为它们被部署在各个领域。这包括物流、运输、农业、医疗保健、军事行动以及危险环境中的数据收集 [?], [?], [?] 和 [?]。近年来，移动机器人的市场需求显著增加，

原因是应用领域的扩大，机器人技术的进步以及电子传感器和设备成本的下降。因此，为了满足这些需求并探索新市场，全球各地的公司正在开发移动机器人。

导航在移动机器人设计中起着基础性作用，使其能够在多样且通常具有挑战性的环境中精确确定位置和方向。在室外操作时，导航任务依赖于多种传感器，包括相机 [?]、LiDAR [?]、声呐 [?]、全球导航卫星系统 (GNSS) [?]、惯性导航系统 (INS) [?] 和里程计 [?]。在室内或隧道中，GNSS 信号不可用，而视觉方法则因光线不足或缺乏特征的环境而受限。在这样的现实场景中，导航解决方案仅依赖于惯性传感器读数，这一过程被称为纯惯性导航 [?], [?]。

那里，由于惯性测量中的误差和噪声，导航解会漂移。为了应对惯性漂移，Shurin 和 Klein 提出了四旋翼的推算方法。受到行人自然运动的启发，他们在四旋翼上执行周期性运动并设计了一种基于模型的定位方法。后来，类似的基于模型的方法也应用于移动机器人。随着深度学习方法在导航领域的出现，设计出了用于四旋翼和移动机器人的神经网络解决方案。

近年来科学机器学习的进步，为此类挑战提供了有前景的解决方案，并揭示了新的科学现象，特别是通过物理信息神经网络 (PINNs)。与传统的深度神经网络 (DNNs) 不同，PINNs 将通常以微分方程 (DEs) 表达的物理定律直接嵌入神经网络的目标函数中，作为残差损失 [?]。这种整合确保了模型输出遵循主要的物理原则，提高了准确性和可解释性，同时减少了对大量标记数据的依赖。PINNs 利用并行计算和自动微分，以解决反复求解微分方程的计算效率问题。在过去几年中，PINNs 在科学和工程领域表现出了显著的成功，包括异常检测 [?], [?]、流体力学 [?]、固体力学 [?]、遥感 [?]、机器人操作 [?], [?] 等。这些进步突出了 PINNs 作为一种强大的工具，可用于建模复杂的动态系统，尤其是在传统方法因数据噪声而难以应对的情况下。在导航和控制领域，Xu 等人 [?] 开发了一种基于 PINN 的模型，用于预测无人水面车辆的横荡和纵荡速度以及旋转速度。在最近的工作中，Park 和 Lee [?] 采用 PINN 算法和误差跟踪控制来估算领航机器人速度。

在 PINN (物理信息神经网络) 的进展基础上，本文介绍了 MoRPI-PINN，一种用于精确导航的移动机器人纯惯性框架。假设 PINNs 可以通过普通或偏微分方程以及有限数据的形式使用部分物理知识有效地指导模型。PINNs 通过将已知的物理定律（如此处的二维惯性导航系统方程）直接嵌入到神经网络训练中，为此类场景提供了一个统一的框架。这是通过自动微分实现的，它在损失函数中用作软约束来强制执行支配方程。在惯性导航中，MoRPI-PINN 模型通过平衡物理一致性和数据保真度来学习系统的轨迹。这使得它们特别适合于有限和/或噪声传感器数据的导航任务。为了进一步增强 MoRPI-PINN 框架的影响力，我们限制机器人作蛇形滑动运动，因为这种运动已被证明可

A. K. S. is supported in part by the Maurice Hatter Foundation. (Corresponding author: Arup Kumar Sahoo)

The authors are with the Autonomous Navigation and Sensor Fusion Lab (ANSFL), the Hatter Department of Marine Technologies, Charney School of Marine Sciences, University of Haifa, Haifa 3498838, Israel. (e-mail: asahoo@campus.haifa.ac.il, kitzik@univ.haifa.ac.il)

以提高惯性信号的信噪比，允许回归移动机器人的位置，即使在崎岖地形中。

本文的贡献包括：

- 开发 MoRPI-PINN 框架以应对移动机器人在各种场景中操作的纯惯性导航的真实场景。
- 在网络的训练过程中，仅依靠单一轨迹，将二维不可压缩流动方程的控制物理与稀疏传感器数据相结合。

我们证明了通过将 2D-INS 运动方程的物理定律和约束嵌入到训练过程中，MoRPI-PINN 为移动机器人提供了准确且稳定的导航解决方案。利用在配备了 IMU 和 RTK-GNSS 的移动机器人上的真实实验，我们展示了与其他基于模型和数据驱动的方法相比提高了 85 % 的性能。

接下来的论文结构如下：第 II 节介绍了 INS 方程及各种基于模型的解决方案。第 ?? 节提出了我们的方法，详细说明了 PINNs 算法的构建和模型架构。第 ?? 节描述了我们的实验设置并给出了全面的结果，以及数据分析。最后，第 ?? 节给出结论。

II. 基于模型的方法

本节简要概述了基于 2D-INS 模型的解决方案和 MoRPI-PINN 预测，这些已被用作基准，以便与我们提出的 MoRPI-PINN 方法进行比较。

A. 惯性导航系统

惯性导航系统提供了全面的导航解决方案，包括位置、速度和方位。此外，基本的 INS 运动方程通常是在导航坐标系 (n 坐标系) 中使用东北天 (NED) 坐标编写的。但对于实际的移动机器人导航，通常使用一个更为方便的局部坐标系 (l 坐标系)。该 l 坐标系在机器人起始位置处定义，坐标轴与北、东、下方向对齐。INS 运动方程如下所示：[?]

$$\mathbf{p}^n = \mathbf{v}^n, \quad (1)$$

$$\mathbf{v}^n = \mathbf{C}_b^n \mathbf{f}_{ib}^b \mathbf{g}^n, \quad (2)$$

$$\mathbf{C}_b^n = \mathbf{C}_b^n \mathbf{\Omega}_{ib}^b. \quad (3)$$

其中在 (1) – (3) 中， $\mathbf{p}^n$ 表示 l 框架中的位置向量， $\mathbf{v}^n$ 代表相同框架中的速度向量， $\mathbf{g}^n$ 是重力向量，假设在整个轨迹中保持不变。矩阵 $\mathbf{C}_b^n$ 是一个正交旋转矩阵，将向量从机体框架 (b 框架) 转换到 n 框架。特定力 $\mathbf{f}_{ib}^b$ 由加速度计在 b 框架中测量，而 $\mathbf{\Omega}_{ib}^b$ 是一个反对称矩阵，表示由陀螺仪测量的角速度。

加速度计测量 b-框架中的特定力，由向量表示：

$$\mathbf{f}_{ib}^b = [f_x \ f_y \ f_z]^T, \quad (4)$$

其中 f_x, f_y, f_z 是沿 b-框架轴的加速度计读数。

类似地，陀螺仪测量 b 坐标系中的角速度为：

$$\mathbf{\Omega}_{ib}^b = [\omega_x \ \omega_y \ \omega_z]^T. \quad (5)$$

。这里， $\omega_x, \omega_y, \omega_z$ 是角速度向量的分量。下标 ib 表示相对于 l 坐标系的测量，而上标 b 表示在 b 坐标系内的分辨率。反对称角速度矩阵 $\mathbf{\Omega}_{ib}^b$ 定义为：

$$\mathbf{\Omega}_{ib}^b = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix}, \quad (6)$$

，其中， $\omega_x, \omega_y, \omega_z$ 是角速度向量 $\mathbf{\Omega}_{ib}^b$ 的分量。

对于轮式机器人运动，通常假设滚动和俯仰角可以忽略不计，运动主要发生在 $x - y$ 平面内。在这个假设下，体坐标系到导航坐标系的变换仅取决于偏航角 ψ ，并由 [?] 给出：

将 (??) 代入 (2) 表明，在水平平面内 f_z 不影响位置或速度。因此，在惯性计算中可以排除它。此外，由于只有偏航角相关，陀螺仪组件 ω_x 和 ω_y 被忽略，只保留 ω_z 。此外，由于使用低成本惯性传感器和涉及的时间间隔相对较短，地球的自转速率和运输速率在 (2)–(3) 中也被忽略。

B. MoRPINet

MoRPINet 是一种最新的、专门为移动机器人纯惯性定位方法设计的数据驱动模型。它利用移动机器人的蛇形动态运动，并遵循三阶段过程进行航位推算位置更新。它可以列为：

值得一提的是，MoRPINet 不提供速度信息。

在这项工作中，我们提出了一种新的纯惯性定位框架 MoRPI-PINN，该框架专为在周期运动中移动的轮式机器人设计。MoRPI-PINN 利用此类机器人的强制机动性来学习周期性轨迹模式。一个设计的误差状态观察器将二维惯性导航系统 (2D-INS) 的运动方程作为物理约束整合到模型中，以提高预测准确性。MoRPI-PINN 框架将这些已知的物理约束与传感器数据融合，以准确跟踪移动机器人的轨迹。

C. MoRPI-PINN 损失函数

PINN 损失函数的构建和优化对于任务的成功至关重要。在我们 MoRPI-PINN 方法的训练过程中，主要目标是优化网络参数 (权重和偏差)，以同时满足观察到的惯性传感器数据和由二维 INS 动力学强加的基本物理约束。

我们提出的神经网络由参数 θ 表示，该参数代表网络中所有可训练的权重和偏差。在训练期间，优化 θ 以最小化总损失函数，记为 $\mathcal{L}$ ，该函数结合了来自加速度计和陀螺仪测量的监督损失 $\mathcal{L}_{data}$ ，以及来自二维惯性导航系统运动方程的无监督物理损失 $\mathcal{L}_{phys}$ ，还有初始条件损失 $\mathcal{L}_{init}$ 。每个子损失项的详细描述如下。

- **Data Loss:** 惩罚预测状态 (位置和速度) 与其 GT 对应值之间的差异。数据损失被计算为网络预测与移动机器人 2D 位置和速度向量的 GT 值之间的均方误差 (MSE)。

$$\mathcal{L}_{data} = \frac{1}{N_{data}} \sum_{i=1}^{N_{data}} (\|\mathbf{p}_i - \hat{\mathbf{p}}_i\|^2 + \|\mathbf{v}_i - \hat{\mathbf{v}}_i\|^2), \quad (7)$$

其中 $\mathbf{p}_i$ 和 $\mathbf{v}_i$ 分别表示数据点 i 处移动机器人的 2D 位置和速度向量的 GT 值，而 $\hat{\mathbf{p}}_i$ 和 $\hat{\mathbf{v}}_i$ 则表示从网络获得的相应预测向量。术语 N_{data} 指的是有 GT 测量值可用的监督训练样本数量。

- **初始条件损失:** 为了确保在每个输入窗口的开始进行准确的轨迹估计，引入了初始条件损失。该损失对预测的初始位置与其已知初始值之间的差异进行惩罚。其计算方式如下：

$$\mathcal{L}_{init} = \frac{1}{N_{ic}} \sum_{i=1}^{N_{ic}} (\|\hat{\mathbf{p}}_i - \mathbf{p}_0\|^2), \quad (8)$$

其中 $\mathbf{p}_0$ 是初始位置矢量， N_{ic} 是用于加强初始条件损失的样本数量。在此，我们设定 $N_{ic} = 200$ ，对应于

训练期间使用的 200 个初始条件点。这些点是通过重复训练数据中的第一次观测传感器读数生成的。

- 物理损失：除了监督学习损失外，还有一个物理信息损失被加入到模型中，以确保网络的预测与基础的 2D-INS 动力学一致。物理信息残差损失函数计算如下：

$$\text{phys} = \frac{1}{N_{\text{phys}}} \sum_{i=1}^{N_{\text{phys}}} \left(\left\| \frac{d\hat{\mathbf{p}}^{n t_i}; \theta}{dt} - \hat{\mathbf{v}}^{n t_i}; \theta \right\|^2 + \left\| \frac{d\hat{\mathbf{v}}^{n t_i}; \theta}{dt} - \left(\hat{\mathbf{C}}_b^{n t_i}; \theta \mathbf{f}_{ib}^{b t_i} \mathbf{g}^{n t_i} \right) \right\|^2 + \left\| \frac{d\hat{\mathbf{C}}_b^{n t_i}; \theta}{dt} - \hat{\mathbf{C}}_b^{n t_i}; \theta \mathbf{\Omega}_{ib}^{b t_i} \right\|_F^2 \right), \quad (9)$$

其中， N_{phys} 是用于通过物理损失来强制执行基础物理定律的插值点的数量。在此，我们生成一组 $N_{\text{phys}} = 2000$ 个插值点，这些点在时间域上均匀采样，并在输入传感器空间内随机采样。这样可以确保物理损失不仅在观测到的数据点处进行评估，还贯穿于更广泛的时空域中；以实现 2D-INS 动力学的泛化和一致性。此外， θ 表示神经网络的可训练参数集，符号 $\|\cdot\|_F$ 表示矩阵的 Frobenius 范数。该损失计算了基于物理模型 (1) – (3) 给出的网络预测动力学与状态的真实时间导数之间的差异。物理损失可明确表示为：

$$\text{phys} = \frac{1}{N_{\text{phys}}} \sum_{i=1}^{N_{\text{phys}}} \left[\left\| \frac{d}{dt} \begin{bmatrix} \hat{x} t_i \\ \hat{y} t_i \end{bmatrix} - \begin{bmatrix} \hat{v}_x t_i \\ \hat{v}_y t_i \end{bmatrix} \right\|^2 + \left\| \frac{d}{dt} \begin{bmatrix} \hat{v}_x t_i \\ \hat{v}_y t_i \end{bmatrix} - \left(\hat{\mathbf{C}}_b^{n \hat{\psi} t_i} \begin{bmatrix} f_x t_i \\ f_y t_i \end{bmatrix} \begin{bmatrix} g_x \\ g_y \end{bmatrix} \right) \right\|^2 + \left(\frac{d\hat{\psi} t_i}{dt} - \omega_z t_i \right)^2 \right], \quad (10)$$

其中， $\hat{x} t_i$ 和 $\hat{y} t_i$ 分别是预测的位置分量， $\hat{v}_x t_i$ 和 $\hat{v}_y t_i$ 分别是预测的速度分量，而 $\hat{\psi} t_i$ 是预测的偏航角。2D 主体到导航的旋转矩阵 $\hat{\mathbf{C}}_b^{n \hat{\psi}}$ 表示为：

$$\hat{\mathbf{C}}_b^{n \hat{\psi}} = \begin{bmatrix} \cos \hat{\psi} & -\sin \hat{\psi} \\ \sin \hat{\psi} & \cos \hat{\psi} \end{bmatrix}.$$

我们的 MoRPI-PINN 模型的总损失是上述三个组成部分 (7)、(8) 和 (10) 的加权和：

$$\text{total} = \lambda_{\text{data}} \lambda_{\text{init}} \lambda_{\text{phys}}, \quad (11)$$

其中 λ_{data} 、 λ_{init} 和 λ_{phys} 是控制每个损失组件相对重要性的超参数。这些权重经过调整，以确保在神经网络的训练过程中充分体现经验测量和物理定律。

D. 网络架构

MoRPI-PINN 模型的架构被设计用于移动机器人状态估计，以捕捉导航的复杂动力学。为了处理传感器数据并近似一个控制方程系统，专门设计了 DNN。其架构由 10 个隐藏层组成，每层有 128 个神经元。我们在每一层应用层归一化，使用非线性激活函数 SinTanh [?]。激活函数 $\phi \cdot$ 定义为：

$$\phi z_i^\ell = \sin z_i^\ell \cdot \tanh z_i^\ell, \quad (12)$$

，其中 z 是由以下定义的输出层：

$$z_i^\ell = \sum_{j=1}^{n_{\ell-1}} \omega_{ij}^\ell a_j^{\ell-1} b_i^\ell, \quad (13)$$

，其中 $\theta = \{\omega^\ell, b^\ell\}$ 表示可训练参数的集合， ℓ 是当前层的索引， L 是神经网络中层的总数， n_ℓ 表示在 ℓ^{th} 层中的神经元数量。

图 ?? 展示了我们提出的 MoRPI-PINN 框架在移动机器人纯惯性导航中的架构。网络接收时间 (t)、来自加速度计的特定力测量 (f_x, f_y) 以及来自陀螺仪的角速度 (ω_z) 作为输入。这些通过一个全连接神经网络处理，以预测机器人的二维位置 ($\hat{x}, \hat{y}$)、速度 ($\hat{v}_x, \hat{v}_y$) 和偏航角度 ($\hat{\psi}$)，统称为 $\hat{\mathbf{N}}$ 。

由于 PINNs 方法是无网格的，因此使用了分散的训练点而不是类似格子的结构点。这为训练复杂轨迹提供了更多的灵活性。该模型使用批大小为 32 和初始学习率为 0.001 训练了 1,000 个周期。为增强泛化能力和防止过拟合，使用了丢弃法，率为 0.2。若验证损失在 50 个周期内趋于平稳，学习率调度器按 0.5 的因子减少学习率 (η)，以进行广泛的初始调整。采用了早停法来防止过拟合。惯性数据使用 10 个样本的重叠窗口进行分段，相邻段之间重叠 12 个样本，以保持时间一致性。

在训练过程中，网络参数 θ 使用 Adam 优化器进行优化，该优化器以其在处理复杂优化挑战中的有效性而闻名。优化器结合权重衰减 ($1e-04$) 使用，以提供 $L2$ 正则化。通过 PINNs 框架得出估计状态 ($\hat{\mathbf{N}} t$) 及其相对于可训练参数的导数。用于训练 MoRPI-PINN 模型的超参数汇总于表格 ?? 中。用于训练模型的硬件配置汇总于表格 I 中。MoRPI-PINN 训练过程概述于算法 1 中。

TABLE I
硬件设置：模型训练用的 GPU 规格

Parameter	Details
GPU Model	NVIDIA GeForce RTX 4090
CUDA Version	12.6
Driver Version	560.35.05
Memory Size	24 GB
Tensor Cores	512
Compute Capability	8.9
Operating System	macOS 15.4.1

Algorithm 1 MoRPI-PINN 训练过程

```

1: Input: Initial weights  $\mathbf{w}_0$ , biases  $\mathbf{b}_0$ 
2: Given:
   • 监督数据  $\{t_i, \mathbf{x}_{\text{true}} t_i\}_{i=1}^{N_{\text{data}}}$ 
   • 初始状态  $\mathbf{x}_0$  在  $t = 0$ 
   • 配置点  $\{t_j\}_{j=1}^{N_{\text{phys}}}$ 
3: Initialize: Iteration counter  $k \leftarrow 0$ , convergence threshold  $\epsilon$ 
4: while not converged do
5:   Predict:  $\hat{\mathbf{x}} t$  using current network parameters
6:   Compute total loss:  $\text{total}$  according to (11)
7:   Update network parameters using suitable optimizer
8:   if change in  $\text{total} < \epsilon$  then
9:     break
10:  end if
11:   $k \leftarrow k + 1$ 
12: end while

```

我们首先描述我们分析中使用的数据集，随后给出评估矩阵的定义。然后，我们对所提出的方法进行了详细分析，

并将其与模型驱动的基线 (2D-INS) 和数据驱动的基线 (MoRPI-Net) 进行比较。

实验 GT 数据是通过 Etzion 等人进行的田野实验收集的。在这个实验中,使用了一辆 RC 车,型号为 STORM Electric 4WD Climbing Car。该车的尺寸为 $385 \times 260 \times 205$ mm,轴距为 253 mm,轮胎直径为 110 mm。RC 车装配了 Javad SIGMA-3N RTK GNSS 传感器,提供高精度的定位测量,其精度为 10 cm,采样率为 10 Hz,作为 GT 使用。此外,RC 车还安装了 Movella DOT IMUs,可以在 120 Hz 运行。关于数据集的更多细节在 [?] 中提供。由于 IMU 的工作频率为 120 Hz,而 RTK 的频率为 10 Hz,因此一个 RTK 样本对应于十二个 IMU 样本。加速度计和陀螺仪的相关噪声和偏差值见表 ??。

训练轨迹如图 ?? 所示,持续时间为 794 秒,是训练的主要数据。

测试数据集由相同实验研究 [?] 中的四个轨迹组成,共覆盖约 25 米。这些轨迹包括直线和周期性运动模式。四个测试轨迹如图 ?? 所示,分别记录为 37、46、39 和 70 秒,总计 192 秒的惯性数据。相同的数据集用于评估基线方法和我们提出的 MoRPI-PINN 框架。

在训练中总共结合了 200 个初始条件点。这些通过复制训练轨迹 (图 ??) 中首次观测到的传感器读数合成生成,以在配点样本中提供稳定的初始化。此外,在训练和测试之前,数据经过以下预处理管道:

对于我们的分析,我们使用以下评估指标:

在深入探讨结果之前,我们注意到对于移动机器人定位性能,我们将我们的方法与基线方法进行比较。然而,对于速度比较,只考虑 2D-INS 模型,因为 MoRPI-Net 不提供速度信息。2D-INS 使用 (1) - (3) 计算,而 MoRPI-Net 在相同条件下进行评估以进行比较分析。为提高基线准确性,通过利用每条轨迹的初始静止段,对加速度计和陀螺仪数据应用了零阶校准。相比之下,提出的 MoRPI-PINN 模型和 MoRPI-Net 使用原始传感器数据,以展示其在噪声条件下有效操作的能力。对这三种方法的评估是在测试数据集上进行的,该数据集由四种不同的轨迹组成:直线路径 (图 ??),周期性轨迹 (图 ?? - ??),以及 L 形路径 (图 ??)。

1) 二维-INS: 对于 2D-INS 解决方案进行了评估,因为它是导航系统中最广泛采用的基线模型之一。如预期的那样,它产生了显著的误差。对于位置估计,该模型在测试数据集上的平均 ATE 为 14.3 米。

此外,2D-INS 模型在速度估计方面的准确性显著较差,NVRMSE 在直线轨迹中,对于 v_x 达到 1077 %,对于 v_y 达到 1211 %。在周期性轨迹中,误差依然较高,对于轨迹 2 的 v_x 和 v_y 分别为 51 % 和 1060 %,对于轨迹 3 分别为 37 % 和 2149 %。L 型轨迹带来的挑战进一步放大了这些误差,对 v_x 和 v_y 的 NVRMSE 分别达到 277 % 和 175 %。这些结果突显了基于模型的惯性导航方法在处理动态运动下的长时间持续和急转弯时的局限性。

2) MoRPI-Net: 数据驱动的 MoRPI-Net 作为导航的一个强大基线方法,相较于传统的 2D-INS 提供了显著的定位精度提升。数据驱动的 MoRPI-Net 达到了 85% 的精度,所有轨迹的平均 ATE 为 5.7 米。这展示了深度学习模型通过直接从传感器输入中捕捉数据模式来校正惯性漂移的潜力。

而且,尽管在位置估计方面有所提高,它在涉及突然方向变化的场景中仍然表现不佳,其预测偏差更加显著。

3) MoRPI-PINN: 在所有轨迹中实现了相对较低的平均轨迹误差 (ATE),平均 ATE 为 0.8 米,相比 2D-INS 提高了 94 %,相比 MoRPI-Net 提高了 85 %。

在速度估计方面,MoRPI-PINN 始终表现出较低的 NVRMSEs。对于直线轨迹,它在 v_x 中达到 54 %,在 v_y 中达到 25 %。在周期性运动轨迹中,轨迹 2 的 NVRMSEs 分别为 34 % (v_x) 和 49 % (v_y),轨迹 3 的 NVRMSEs 分别为 37 % (v_x) 和 35 % (v_y)。对于 L 形轨迹,该模型将误差进一步降低到 v_x 中的 20 % 和 v_y 中的 24 %。这些结果强调了 MoRPI-PINN 在多样化轨迹中的稳健性和泛化能力。它展示了在复杂动态环境下,整合二维惯性导航系统运动方程的物理学以实现准确移动机器人导航的优势。

E. 摘要

在所有评估场景中,MoRPI-PINN 框架始终显著优于基线模型。如表 II 所示,MoRPI-PINN 框架将二维惯性导航系统的绝对轨迹误差改善了 94 %,并将 MoRPI-Net 改善了 85 %。该模型在所有测试轨迹中也展示了优秀的航向推算准确性,最大绝对轨迹误差为 1.1 米,如图 1 所示。由于在学习过程中引入物理损失项 $\mathcal{L}_{\text{phys}}$ 的约束,最大错误始终保持在较低水平。与传统的二维惯性导航系统算法相比,MoRPI-PINN 仅显示偶尔的较大偏差。因此,模型有效地缓解了长期漂移,这是传统惯性导航方法中的一个重要限制。

此外,低初始条件损失 ($\mathcal{L}_{\text{init}} \approx 10^{-3}$) 可以确保轨迹初始化 ($x_0, y_0 \approx 0$) 的准确性,有助于在轨迹开始时减少绝对轨迹误差 (ATE)。输入归一化可以在不同的加速度计 (f_x, f_y) 和陀螺仪 (ω_z) 数据范围内实现一致的性能,并减少在多种测试场景中的错误。这里,物理通知损失 ($\mathcal{L}_{\text{phys}}$),再次在捕获潜在动态方面发挥关键作用。这种正则化有助于防止对噪声测量的过拟合;这是纯数据驱动神经网络的常见限制。这些结果强调了物理通知学习在显著增强惯性导航方面的潜力,尤其是在传感器校准不切实可行或运动偏离简单线性动力学的环境下。

在现实场景中,无 GNSS 环境和较差的光照条件导致了纯惯性导航,从而导致位置解的快速漂移。为了减轻这种漂移,我们引入了 MoRPI-PINN 框架。我们的方法将 2D-INS 控制方程整合到神经网络的训练过程中。它采用复合损失,结合了数据损失、物理损失和初始条件损失。这种设计即使在传感器误差项和噪声存在的情况下,也能实现稳健的学习。

我们在四个实际的轨迹上评估了 MoRPI-PINN,并与两种现有的基于模型的方法进行了比较:(1) 传统的数值 2D-INS 解决方案,以及 (2) 数据驱动的 MoRPI-Net 方法。MoRPI-PINN 实现了仅 0.8 米的位置误差,显著优于 2D-INS (14.3 米) 和 MoRPI-Net (5.7 米)。这相当于对 2D-INS 解决方案提高了 94

总体而言,所提出的模型非常适合多种真实世界的移动机器人应用,包括制造和物流、安全和监控、送货服务以及基础设施检查。MoRPI-PINN 虽然轻量,但具有有效的架构,使其非常适合在嵌入式平台上部署,从而在对抗环境中实现实时导航。因此,MoRPI-PINN 有效地弥合了纯惯性导航的局限性,并在短时间内通过惯性传感器数据实现机器人顺畅导航。

A. K. S. 得到了 Maurice Hatter 基金会的支持。

TABLE II
四条轨迹上的绝对轨迹误差 (米), 平均值和改进

Approach	ATE [m]				Average	Improvement
	Trajectory 1	Trajectory 2	Trajectory 3	Trajectory 4		
2D-INS (model-based baseline)	11.2	7.1	11.3	27.9	14.3	94 %
MoRPINet (data-driven baseline)	5.8	5.1	5.4	6.6	5.7	85 %
MoRPI-PINN (ours)	1.1	0.9	0.9	0.4	0.8	—

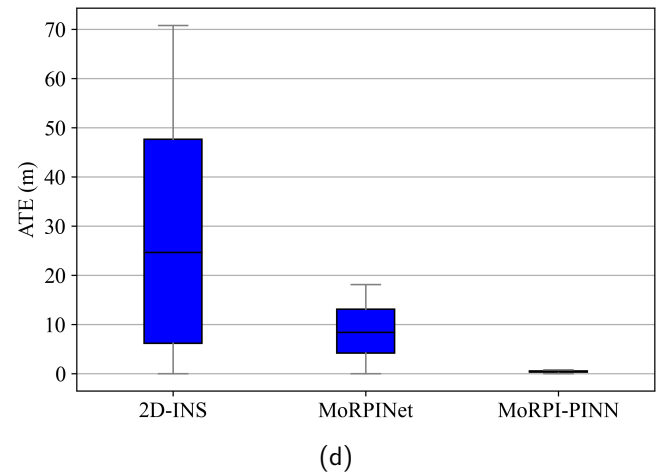
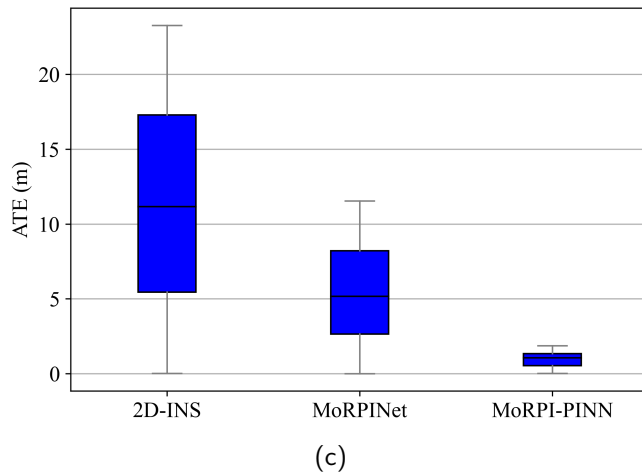
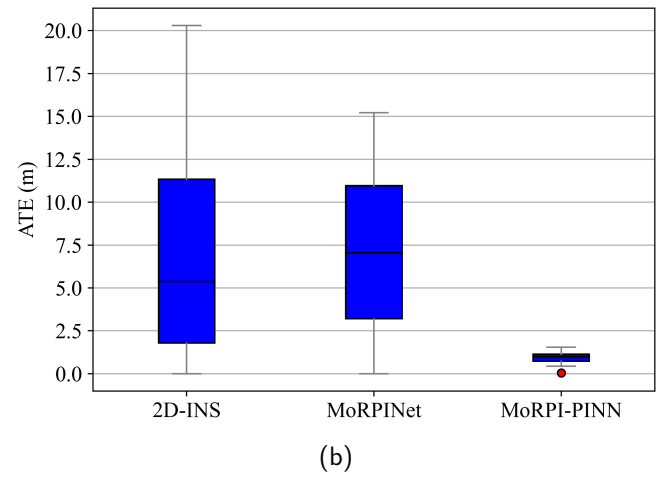
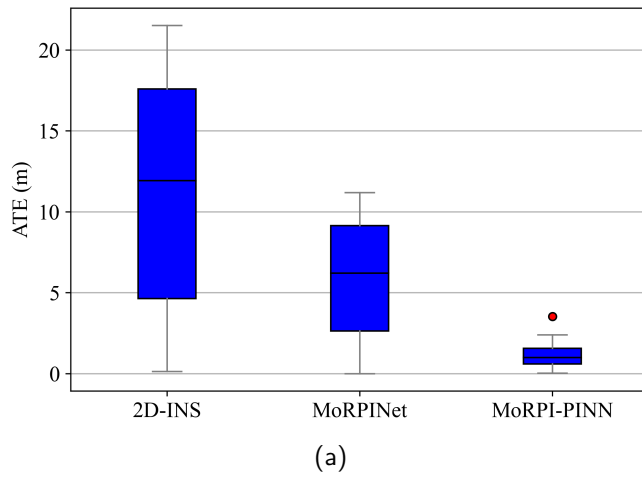


Fig. 1. 模型基础基线 (2D-INS)、数据驱动基线 (MoRPINet) 和我们的 MoRPI-PINN 方法在四个测试轨迹上的 ATE。