# 用于车辆速度轨迹生成的物理信息扩散模型

Vadim Sokolov
乔治梅森大学

Farnaz Behnia
乔治梅森大学

Dominik Karbowski
阿贡国家实验室

arXiv:2602.05028v1 [stat.AP] 4 Feb 2026

摘要—**合成车辆速度轨迹对于评估车辆控制算法和网联车辆技术至关重要。传统的马尔可夫链方法存在离散化伪影和表达能力有限的问题。本文提出了一种用于条件微行程合成的物理信息扩散框架，将双通道速度-加速度表示与软物理约束相结合，从而解决硬约束形式固有的优化冲突。我们使用 6,367 个 GPS 衍生微行程，将一维 U-Net 架构与基于 Transformer 的条件评分扩散插补（CSDI）模型进行了比较。CSDI 实现了更优的分布匹配效果（速度的瓦瑟斯坦距离为 0.30，加速度为 0.026）、与真实数据的高度不可区分性（判别分数为 0.49），并验证了其在下游能耗评估任务中的效用。该方法论无需昂贵的实地数据采集，即可为智能交通系统（ITS）应用实现逼真的驾驶特征曲线的可扩展生成。**

## I. 引言

准确地对车辆速度轨迹建模，是智能交通系统（ITS）中众多应用的基础，包括能耗评估（Karbowski、Sokolov 和 Rousseau，2015 年；Karbowski 等人，2016 年；Karbowski, Sokolov, and Jongryeol，2016 年）、路径优化（Moawad 等人，2021 年）、交通仿真（Auld 等人，2013 年；Auld、Hope 等人，2016 年）以及自动驾驶车辆的预测控制（Dauner 等人，2023 年）。生成能够忠实再现真实世界驾驶行为统计和物理特征的合成速度特征曲线，有助于大规模评估交通政策、车辆技术和交通管理策略，而无需承担大规模实地数据采集所需的高昂成本。

传统的合成轨迹生成方法依赖马尔可夫链模型（Karbowski 等人，n.d., 2016），这类模型将速度状态空间离散化，并对连续状态之间的转移进行建模。虽然这些方法计算效率高且易于解释，但存在根本局限：僵化的离散化会丢失细粒度动态，马尔可夫假设会忽略长期时间依赖关系，而纳入基于物理的约束或条件控制（例如动力总成类型、道路条件）则需要临时设计的工程方案。更多近期研究探索了深度生成模型，包括生成对抗网络（GAN）（Behnia, Karbowski, and Sokolov 2023；Yoon, Jarrett, and van der Schaar2019）和归一化流（Papamakarios 等人 2019），但由于训练不稳定、模式崩溃以及难以施加硬边界约束，效果参差不齐。

本文介绍将去噪扩散概率模型（DDPM）（Ho、Jain和Abbeel，2020年）应用于条件车辆速度轨迹合成问题。扩散模型已成为一类有效的生成模型，在图像生成、音频合成以及近期的时间序列预测中均表现出优异性能（Ansari 等人，2025）。其主要优势包括训练动态稳定、能够自然地建模复杂的多模态分布；对于本应用而言，另一个相关优势是在反向扩散过程中通过修复施加约束的原理性机制。不同于以往将基于扩散模型的轨迹生成用于自动驾驶的研究（Feng 等人，2023 年；Suo等，2021年），这类研究着重于多智能体场景中的空间路径规划，而我们关注用于能耗评估应用的单变量速度曲线，需要严格满足微行程边界条件（初始和最终速度均为零），并精确控制汇总统计量（平均速度、持续时间）。

我们比较了两种基于扩散模型的架构：标准的一维 U-Net 扩散模型，以及由时间序列插补方法改编而来的条件分数扩散插补（CSDI）模型（Tashiro 等，2021年）。两个模型均以行程特征（目标平均速度、持续时间）为条件，并使用来自2007年芝加哥大都会规划署（CMAP）区域家庭出行调查的6,367条微行程观测数据进行训练。我们的研究通过多轮设计迭代，系统地探索了这些模型的演进过程，记录了在标准扩散模型中使用硬约束物理惩罚项所导致的失败，以及在 CSDI 中成功整合软物理约束的情况。我们将这些方法与传统的马尔可夫链基线方法进行基准比较，并分析了其他深度生成方法（DoppelGANger、SDV）的失效模式，为从业者提供指导。

高保真合成轨迹生成在智能交通系统的多个关键领域具有实际效用。在能耗评估方面，合成轨迹有助于在无需昂贵实地试验的情况下，评估电动汽车续航里程、充电基础设施需求，以及适用于多种驾驶场景的车队电动化策略（Chen 等，2016；X. Huang 等，2020）。POLARIS 等交通微观仿真平台（Auld、Hope 等，2016）需要逼真的速度曲线，以准确模拟排放、燃料消耗及网络层面的能耗影响。网联与自动驾驶汽车测试需要多样且符合物理规律的轨迹，以验证感知、规划和控制系统（Mozaffari 等，2020），并生成具有代表性的安全关键场景（Wu 等，2025）。本研究通过展示分布保真度（采用 Wasserstein 度量评估）和下游效用（采用 TSTR 评估），提供了一套经过验证的方法论，用于生成合成驾驶数据，直接支持智能交通系统能源管理、基于仿真的政策分析和智能车辆开发。

本研究的主要贡献如下：

1. 首次将扩散模型专门应用于交通能耗评估中的车辆速度微行程生成；与马尔可夫链和生成对抗网络相比，展现出更优的分布匹配效果。2. 首次成功将运动学物理约束融入面向交通领域的基于Transformer的扩散模型，通过结合速度和加速度直接通道与基于物理的训练目标，有效弥合深度生成建模与运动学一致性之间的差距。3. 记录了完整的模型开发生命周期，包括失败的尝试（硬约束扩散模型、DoppelGANger、SDV），揭示某些方法失败的原因，以及如何避免交通应用中生成建模的常见问题。4. 建立了严谨的评估框架，涵盖分布保真度（瓦瑟斯坦距离、MMD）、运动学有效性（平滑度指标、边界违规）和效用（判别分数、TSTR），为评估合成轨迹质量确立最佳实践。5. 在 https://github.com/VadimSokolov/diffusion-trajectory-generation提供开源实现和可复现性套件，其中包括完整的训练代码、预训练模型权重、预处理数据和图表生成脚本，从而能够完整复现所有结果。

本文其余部分组织如下：第II 节回顾车辆轨迹生成、时间序列深度生成模型以及扩散模型的相关工作。第III 节介绍 CMAP 数据集、预处理和聚类分析。第IV 节阐述方法论，包括问题表述、马尔可夫链基线，以及扩散模型和 CSDI 架构的详细描述，并说明它们在多次设计迭代中的演进。第V 节报告定量与定性结果，比较所有方法并记录失效模式。第VI 节讨论对实践的启示、局限性以及未来研究方向。

## II. 相关工作

交通领域的研究从多个角度探讨了如何生成逼真的车辆轨迹。早期研究采用参数化模型和马尔可夫过程来捕捉驾驶模式（Karbowski 等人，n.d., 2016），将速度转移建模为离散状态随机过程。Karbowski 等人开发了用于节能车辆路径规划中行程预测的马尔可夫链模型，并表明二阶马尔可夫模型能够捕捉速度与加速度之间的依赖关系（Karbowski, Sokolov, and Jongryeol 2016）。这些方法虽然具有可解释性且计算效率高，但受限于其无记忆特性，无法对驾驶行为中固有的长程时间依赖关系进行建模。

POLARIS 等基于智能体的交通仿真平台（Sokolov, Auld, and Hope 2012; Auld et al. 2013; Auld, Hope, et al. 2016）通过对单个车辆运动进行微观仿真来评估区域交通政策。这些系统将基于活动的需求模型与网络分配相结合，需要以逼真的速度曲线作为输入（Auld, Karbowski, et al. 2016）。Moawad 等人提出了用于实时路径分配的神经推荐系统，并将能耗纳入考量，凸显了根据路径特征快速生成具有代表性的速度轨迹的必要性（Moawad 等人，2021）。

最近，Behnia 等人探索了用于车辆速度轨迹的深度生成模型，比较了变分自编码器（VAEs）、生成对抗网络和归一化流（Behnia, Karbowski, and Sokolov 2023）。他们的分析表明，生成对抗网络在分布匹配方面表现最佳，但存在训练不稳定和模式崩溃问题，尤其是在多样化的驾驶工况下。该研究为轨迹生成确立了重要的评估指标，但未探索扩散模型；此后，扩散模型已成为比生成对抗网络更稳定的替代方案。

与单车速度生成相邻的研究方向，是针对自动驾驶车辆测试的多智能体轨迹预测和场景生成，其相关文献正在迅速增长。TrafficGen（Feng等，2023年）和ScenarioNet（Li等，2023年）提出了基于学习的框架，可从大规模数据中生成多样且逼真的交通场景

集。TrafficSim（Suo等，2021年）提出了一种多智能体行为模型学习方法，用于模拟车辆之间逼真的交互。

这些研究与我们的关注点存在根本差异：它们生成的是二维道路网络中多个交互主体的空间轨迹（随时间变化的 x、y 坐标），而我们关注的是独立微行程的单变量速度曲线。前者需要对复杂的空间交互和碰撞规避进行建模；后者则侧重于匹配运动学分布，并施加严格的边界条件以进行能耗评估。尽管如此，扩散模型近期在自动驾驶轨迹规划（Liao等，2024年）和可控运动生成（Lan等，2025年）中的应用，展现了这种生成建模范式在交通运输应用中的潜力。

深度生成模型在序列数据上的应用发展迅速。TimeGAN（Yoon, Jarrett, and van der Schaar，2019年）通过结合无监督对抗学习与有监督的逐步预测，将生成对抗网络拓展至时间序列领域，并提升了时间一致性。DoppelGANger通过双阶段生成过程，解决了生成兼具时间特征和静态属性的混合类型时间序列这一难题。然而，这两种方法仍容易出现模式崩溃，且需要仔细调整架构才能施加约束。

自回归模型，包括 Synthetic Data Vault (SDV) 中实现的概率变体，是另一种常见方法。PARSynthesizer 使用概率自回归，根据先前的值依次生成各个时间步。虽然这类模型易于实现，但它们存在曝光偏差——错误会在长序列中不断累积——而且若没有明确的多样性机制，也难以捕捉多样化的多模态分布。

时间序列基础模型近来已成为零样本替代方案。Chronos（Ansari 等人，2025）通过对时间序列进行标记化，并在多样化的数据集上训练，将 Transformer 语言模型架构应用于概率预测。尽管这类模型在预测任务中展现出潜力，但它们缺乏对生成过程的细粒度控制，也尚未针对受约束的轨迹合成进行广泛评估。

去噪扩散概率模型（DDPM）（Ho、Jain和Abbeel，2020年）定义了一个前向过程，通过高斯噪声逐步扰乱数据，并学习一个反向过程，将随机样本去噪为数据。与生成对抗网络不同，扩散模型优化一个易于处理的变分下界，从而实现稳定训练并获得高质量样本。基于分数的生成模型（Y. Song等，2020年）通过学习

数据对数密度的梯度（分数函数），并使用朗之万动力学进行采样，提供了一种等价的表述。

扩散模型在时间序列中的应用主要集中于预测和插补。CSDI（用于插补的条件得分扩散模型）（Tashiro 等人，2021）提出了一种基于 Transformer 的扩散架构，用于多变量时间序列中的缺失值插补，且表现优于自回归和基于生成对抗网络的方法。其关键创新在于利用自注意力捕捉长程依赖，同时由扩散框架处理不确定性。我们通过将整条轨迹视为“缺失”，并以汇总行程统计数据为条件，对该架构进行改造以实现条件生成。智能交通系统领域近期的扩展工作将扩散模型与生成对抗网络结合用于轨迹重建（Qian 等，2025），并提出了用于速度预测的可解释因果扩散网络（Rong 等，2025）。

TimeGrad（Rasul 等，2021年）提出了用于概率时间序列预测的自回归去噪扩散方法，并在多变量预测任务中展现出优异性能。SSSD（Alcaraz 和 Strodthoff，2022年）将结构化状态空间模型与扩散模型相结合，用于插补和预测。TimeWeaver 和 Diffusion-TS 进一步将扩散模型扩展到多变量时间序列合成领域。然而，这些方法并未解决车辆轨迹生成所面临的特定挑战：严格的边界约束（起始/终止速度为零）、符合物理规律的合理性（加速度限制、平滑度），以及细粒度条件控制（车辆类型、道路状况）。尽管深度学习已广泛应用于智能交通系统中的轨迹预测（Schultz 和 Sokolov，2018年；Mozaffari 等，2020年；Zhao 等，2019年；Altché 和 de La Fortelle，2017年；Y. Huang 等，2022年），但用于能耗评估的完整速度曲线生成仍未得到充分探索。多智能体扩散模型也已用于交通流预测（Adam 等，2025年）。

据我们所知，本研究首次专门将扩散模型应用于用于能耗评估的车辆速度微行程生成，系统比较了 U-Net 与基于 Transformer的架构，并记录了物理信息约束的集成方法。

## III. 数据

本研究的数据来自芝加哥大都会规划署（CMAP）2007年区域家庭出行调查。该调查由 NuStats 代表 CMAP 开展，覆盖一个包含11个县的区域，其中包括高密度城市地区（芝加哥）、郊区和农村环境。这种地理多样性确保了对不同驾驶条件、道路类型和交通模式的充分代表。

该调查采用双框抽样策略，结合随机数字拨号 (RDD) 和基于地址的

抽样，覆盖了超过 320 万个家庭。数据收集采用计算机辅助电话访谈 (CATI)、邮寄问卷和后续电话访谈，最终总体应答率为 10%（招募率为 19%，回收率为 55%）（芝加哥大都会规划署，2008年）。部分参与家庭提供了基于 GPS 的出行层面数据，这些数据构成了本研究分析的基础。

该数据集记录了用于个人出行的乘用车行驶情况（轿车、运动型多用途汽车、厢式车和皮卡）。不包括重型商用车辆和车队运营。道路类型涵盖州际公路和快速路、连接经济中心的主干道，以及城市和郊区的本地道路。数据采集过程中未按道路类型进行筛选；GPS轨迹反映了家庭出行模式的自然分布。

### *A.* 探索性数据分析

在预处理并提取出独立的微行程（连续的速度曲线，起点和终点速度均为零）后，我们的数据集包含 6,367 个以 1 赫兹采样的观测值。表 I 总结了关键的分布统计量。

表 I：数据集汇总统计

| 属性 | Min | Max | Mean | 中位数 |
|---|---|---|---|---|
| 持续时间（秒） | 34 | 12,841 | 304 | 187 |
| 持续时间（分钟） | 0.57 | 214 | 5.06 | 3.12 |
| 距离（米） | 257 | 393,070 | 5,884 | 3,076 |
| 距离（千米） | 0.26 | 393 | 5.88 | 3.08 |
| 平均速度（米/秒） | 5.18 | 31.57 | 16.92 | 16.45 |
| 平均速度（千米/小时） | 18.63 | 113.66 | 60.92 | 59.21 |

该数据集中的行程特征差异显著：行程时长从半分钟到三个多小时不等，距离跨度达两个数量级。平均速度从缓慢的城市拥堵行驶（5 米/秒 $\approx$ 18千米/小时）到高速公路巡航（32 米/秒 $\approx$ 115千米/小时）不等，体现了大都市区驾驶情境的多样性。

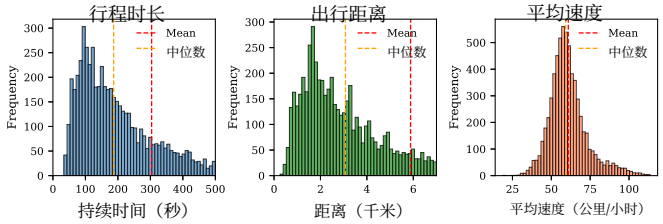



图1：行程时长、距离和平均速度的数据分布。直方图显示了 6,367 次微行程的经验分布，并标出了均值（红色虚线）和中位数（橙色虚线）。这些分布表现出显著的异质性，体现了从城市拥堵到高速公路巡航的多样化驾驶情境。

为刻画数据集中驾驶模式的异质性，我们对从每条轨迹中提取的特征进行了K均值聚类（K{v1}）。聚类有两个目的：（1）了解数据结构，为模型设计提供依据；（2）支持分层评估，确保生成的轨迹涵盖所有驾驶工况。

特征提取：对于每次微行程，我们计算了：

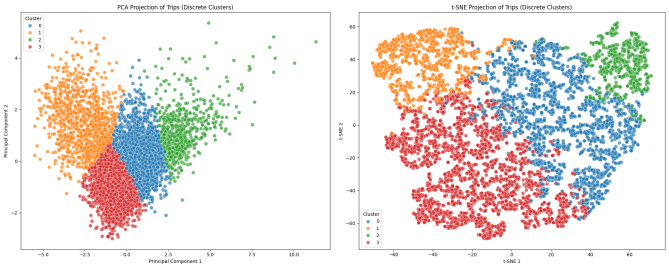
- 平均速度（$\bar{v}$）和最大速度（$v_{max}$）
- 速度标准差
- 怠速时间比（速度为 $v < 0.5$ m/s 时所占的时间比例）
- 每公里停车次数（单位距离内接近零速事件的数量）
- 加速度噪声（加速度的标准差）

聚类结果：算法识别出四种不同的驾驶状态，如表 II所示：

表 II：聚类特征

| 聚类标签 | | 数量 | Avg 速度 （米/秒） | Max 速度 （米/秒） | 每公里停车次数 | Idle 比率 (%) |
|---|---|---|---|---|---|---|
| 0 | 干道/郊区 | 2,224 | 15.6 | 22.6 | 0.59 | 2.0 |
| 1 | 高速公路/州际公路 | 1,020 | 22.2 | 30.8 | 0.12 | 0.5 |
| 2 | 拥堵/City | 636 | 13.9 | 21.8 | 1.29 | 4.4 |
| 3 | 自由流干道 | 2,487 | 16.7 | 22.7 | 0.28 | 1.0 |

簇 1（公路/州际公路）呈现出最高的平均速度和最大速度，停车次数和怠速时间极少，这是不间断高速公路行驶的典型特征。簇 2（拥堵/城市）则呈现相反的特征：速度较低、停车频繁（每公里 1.29 次），且怠速时间较长（4.4%），这是走走停停的城市驾驶的典型表现。簇 0 和簇 3 代表中等程度的干道道路状况，其中簇 3 的速度稍高、停车次数较少，表明干道交通拥堵程度较低。

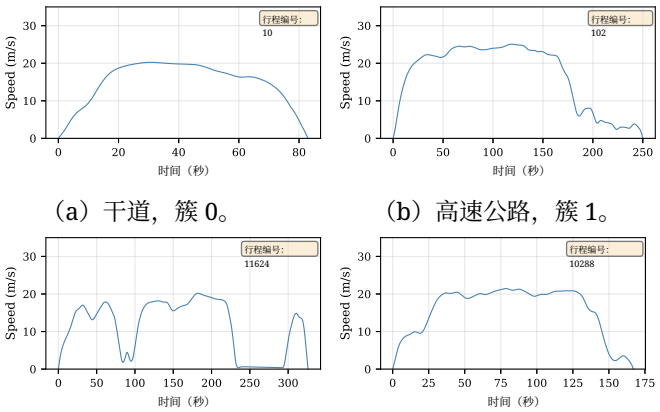



（a）主成分分析　　（b）t-SNE

图 2：特征空间中微行程的主成分分析和 t-SNE 投影。

主成分分析（主成分分析）与 t-SNE 投影（图 2）在特征空间中可视化了簇的分离情况，前两个主成分分析成分解释了约 65% 的方差。t-SNE 揭示了更紧凑的簇结构，凸显了不同的运动学

特征，对应四种驾驶工况。尽管各簇形成了可区分的群体，它们之间的连续过渡反映的是真实世界驾驶条件的连续谱，而非离散类别。与此相关，图 3 展示了每个簇的示例速度曲线，这些曲线选自簇与行程的映射，用以说明典型模式。




（a）干道，簇 0。 （b）高速公路，簇 1。

（c）城市拥堵，簇 2。（d）城市畅通，簇 3。

图3：各簇观测轨迹示例。

这些簇为我们的评估策略提供了依据：我们评估生成的轨迹是否符合各类驾驶工况的比例和特征，确保模型能够捕捉高速公路通行效率和城市拥堵模式。

## IV. 方法论

令$\mathbf{v} = (v_0, v_1, \dots, v_T) \in \mathbb{R}^{T+1}$表示以 1 赫兹采样的车辆速度轨迹，其中 $t \in \{0, 1, \dots, T\}$表示以秒为单位的时间索引。我们的分析单位是微行程：一段始于且止于零速度的停车至停车路段。数据集中的每个完整行程都被划分为一系列此类微行程，即车辆每次停下时便进行分段。这种分解使我们能够对驾驶行为的基本组成部分进行建模；通过串联生成的微行程，可以重建完整的起讫点行程。每个微行程都满足以下约束：

边界条件：

$$v_0 = v_T = 0 \quad \text{(trip begins and ends at rest)}$$

非负性：

$$v_t \geq 0 \quad \forall t \in \{0, \dots, T\}$$

条件变量：我们将条件信息记为$\mathbf{c} \in \mathbb{R}^d$，通常包括：

- $\bar{v}$：目标平均速度 (m/s)
- $T$：行程时长（秒）
- $v_{max}$：最大速度（米/秒） [仅适用于 CSDI]
- $d_{veh}$：车辆动力学参数 $\in [0, 1]$ [仅适用于 CSDI]

目标是学习一个条件生成模型$p_\theta(\mathbf{v}|\mathbf{c})$，该模型：

1. 生成与经验分布 $p_{data}(\mathbf{v})$ 相匹配的轨迹 2. 满足物理合理性约束（平滑加速度、有界加加速度） 3. 通过条件 $\mathbf{c}$ 精确控制汇总行程统计量

形式上，我们旨在最小化分布差异：

$$\min_\theta D_{KL}(p_{data}(\mathbf{v}|\mathbf{c}) \| p_\theta(\mathbf{v}|\mathbf{c}))$$

同时满足边界和运动学约束。

### *A.* 基线：马尔可夫链方法

传统的轨迹生成依赖离散状态马尔可夫模型（Karbowski 等人，未注明日期）。我们遵循 Karbowski 等人的方法论，实现了一个二阶马尔可夫链基线，作为评估深度生成模型的参考。

状态空间离散化：速度按宽度为 $\Delta v = 0.5\,\text{m/s}$ 的区间进行离散化，形成区间 $\{B_0, B_1, \dots, B_K\}$ 其中$B_i = [i\Delta v, (i+1)\Delta v)$。令 $s_t \in \{0, 1, \dots, K\}$ 表示时间 $t$的区间索引。

二阶马尔可夫模型：该模型假设时间 $t$ 的速度取决于前两个时间步：

$$P(s_t | s_{t-1}, s_{t-2}) = \frac{C(s_{t-2}, s_{t-1}, s_t)}{\sum_{s'} C(s_{t-2}, s_{t-1}, s')}$$

其中 $C(\cdot)$ 表示训练数据中的经验计数。该模型可改写为以状态对 $X_t = (s_{t-1}, s_t)$ 为状态的一阶马尔可夫链，其状态空间大小为 $K^2$。

边界约束采样（马尔可夫桥）：为强制满足 $v_0 = v_T = 0$，我们采用前向后向采样（Durham 和 Gallant 2002）。后向消息 $\beta_t(x)$表示从状态 $x$ 在时间 $t$到达终止状态（区间 0）的概率：

$$\beta_T(x) = \mathbb{1}[x_{\text{end}} = 0], \quad \beta_t(x) = \sum_{x'} P(x'|x)\beta_{t+1}(x')$$

每一步的前向采样使用修正后的转移概率：

$$P_{\text{bridge}}(x_{t+1}|x_t) \propto P(x_{t+1}|x_t) \cdot \beta_{t+1}(x_{t+1})$$

后处理：为减少离散化伪影，我们对加速度信号 $a_t = v_{t+1} - v_t$ 进行 5 点移动平均高斯平滑，然后通过积分恢复平滑的速度曲线。

局限性：马尔可夫基线存在三个根本弱点：(1) 僵化的分箱离散化会丢失细粒度动态特征；(2) 有限的时间记忆无法捕捉长距离行程结构（例如先经过高速公路路段，再进行城市导航）；(3) 条件控制需要针对每种条件分别建立模型或采用拒绝采样，这两种方法的效率都很低。

所有模型均使用 PyTorch 2.0 在 NVIDIA A100 GPU（40GB）上训练。训练过程中采用混合精度（FP16）计算，以加快收敛并减少内存占用。马尔可夫链基线无需 GPU 训练，在单个 CPU 核心上不到 5 分钟即可完成参数估计（构建转移矩阵）。

### *B.* 基于一维 *U-Net* 的条件扩散模型

我们的扩散模型采用带有特征级线性调制（FiLM）条件机制的一维 U-Net 编码器-解码器架构。输入是一个双通道张量 $\mathbf{x}_t \in \mathbb{R}^{2\times 512}$，表示联合速度-加速度状态：

$$\mathbf{x}_0 = \begin{bmatrix} v_0, v_1, \dots, v_{511} \\ a_0, a_1, \dots, a_{511} \end{bmatrix}$$

其中，$a_t = v_{t+1} - v_t$ 是离散加速度。短于 512 秒的轨迹会进行零填充；较长的轨迹则会被截断（不足数据的 1%）。

编码器 由四个下采样块组成，通道维度为 $[64, 128, 256, 512]$, ，每个块包含：

- 1D 卷积（核大小 3，步幅 2）
- 两个采用组归一化和 SiLU 激活的 ResNet 块
- FiLM 条件层注入时间嵌入和行程条件

自注意力 层插入在分辨率为 128 和 64 的位置，以捕获长程依赖，同时避免在完整的 512 分辨率下出现计算量激增。

解码器 通过跳跃连接与编码器相对应，并通过上采样重建预测噪声 $\epsilon_\theta(\mathbf{x}_t, t, \mathbf{c}) \in \mathbb{R}^{2\times 512}$。

时间嵌入：扩散时间步 $t \in \{1, \dots, T_{diff}\}$使用正弦位置编码进行嵌入：

$$\mathrm{PE}_i(t) = \begin{cases} \sin(t/10000^{i/d}) & i \text{ even} \\ \cos(t/10000^{i/d}) & i \text{ odd} \end{cases}$$

投影到 256 维。

通过 *FiLM* 进行条件调制：在每个 ResNet 模块中，使用特征级线性调制（FiLM）（Perez 等，2018 年）将条件向量 $\mathbf{c} = [\bar{v}/30, T/1000]$（归一化后）转换为缩放和偏移参数：

$$\mathrm{FiLM}(\mathbf{h}, \mathbf{c}) = \gamma(\mathbf{c}) \odot \mathbf{h} + \beta(\mathbf{c})$$

其中，$\gamma$、$\beta : \mathbb{R}^2 \to \mathbb{R}^{d_h}$ 是通过学习得到的多层感知机投影。这种乘法和加法调制使模型能够根据行程条件调整其特征表示。

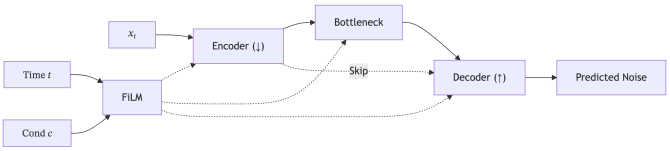



图 4：简化的 U-Net 扩散架构。含噪轨迹 $x_t$ 被编码为潜在表示，再经解码以预测噪声，并由时间 $t$ 和条件$c$进行调制。

*1*）前向扩散过程：我们定义一个前向马尔可夫过程，通过在 $T_{diff} = 1000$ 步内逐渐加入高斯噪声来破坏数据：

$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I})$$

其中，$\bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s)$ 和 $\beta_t$ 按照从 $\beta_1 = 0.0001$ 到 $\beta_{T_{diff}} = 0.02$的线性调度变化。这种选择在噪声破坏（确保 $\mathbf{x}_{T_{diff}} \approx \mathcal{N}(0,\mathbf{I})$）与平滑过渡之间取得了平衡。

*2*）训练目标：模型经过训练，以预测每个扩散步骤中添加的噪声 $\epsilon \sim \mathcal{N}(0,\mathbf{I})$。简化的变分下界目标为：

$$\mathcal{L}_{\text{simple}} = \mathbb{E}_{t\sim U(1,T_{diff}),\mathbf{x}_0\sim p_{data},\epsilon\sim\mathcal{N}(0,\mathbf{I})} \left[\left\|\epsilon - \epsilon_\theta\left(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, t, \mathbf{c}\right)\right\|^2\right]$$

该目标使用 Adam（$\beta_1 = 0.9, \beta_2$ =0.999）进行优化，学习率为 $10^{-4}$，批量大小为 32。

模型使用 $\mathcal{L}_{\text{simple}}$ 训练了 1000 个周期，并通过后处理（尾部渐变和速度缩放）实施边界约束。早期尝试采用硬物理约束（距离匹配、加加速度惩罚、边界约束）均以灾难性失败告终——导致 100% 的边界违规，且分布匹配效果下降（WD 速度 $= 4.40$，而基线为 0.56）。扩散去噪与刚性物理优化之间的这一根本冲突，促使我们转而采用 CSDI 的阈值激活软约束。

*3*）反向采样（生成）：从纯噪声 $\mathbf{x}_{T_{diff}} \sim \mathcal{N}(0,\mathbf{I})$开始，我们使用学习到的反向过程迭代去噪：

$$\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\mathbf{x}_t, t, \mathbf{c})\right) + \sigma_t\mathbf{z}$$

其中，$\mathbf{z} \sim \mathcal{N}(0,\mathbf{I})$，而 $\sigma_t^2 = \frac{(1-\alpha_{t-1})}{(1-\alpha_t)}\beta_t$ 控制反向步骤的随机性。

用于边界约束的修复：在每个反向步骤中，我们通过受约束的修复强制执行 $v_0 = v_T = 0$。定义掩码 $M \in \{0,1\}^{512}$，其中 $M_0 = M_T = 1$

否则为 $M_t = 0$ 。在计算$\mathbf{x}_{t-1}$之后，我们替换被掩码的位置：

$$\mathbf{x}_{t-1} := \mathbf{x}_{t-1} \odot (1 - M) + \mathbf{x}_{t-1}^{\text{known}} \odot M$$

其中$\mathbf{x}_{t-1}^{\text{known}}$在边界索引处为零。这种直接约束确保严格满足约束条件，同时不会削弱已学习到的扩散动态。

无分类器引导 （可选）：为增强条件的作用强度，我们采用尺度为 $w$ 的无分类器引导（Ho 和 Salimans，2022）：

$$\tilde{\epsilon}_\theta = (1 + w)\epsilon_\theta(\mathbf{x}_t, t, \mathbf{c}) - w\epsilon_\theta(\mathbf{x}_t, t, \emptyset)$$

其中 $\emptyset$ 表示无条件生成（通过训练期间以 0.1 的概率随机丢弃条件来实现）。我们将 $w$ 从基线模型中的 1.0 提高到最终模型中的 3.0，以改善平均速度匹配效果。

在 A100 上生成 1000 条轨迹约需 5 分钟。

### *C.* 条件分数扩散插补（*CSDI*）

CSDI 将 Tashiro 等人提出的基于 Transformer 的插补模型（Tashiro 等人，2021）应用于条件轨迹生成。与联合处理速度-加速度的一维 U-Net 不同，CSDI 处理单变量速度序列$\mathbf{v} \in \mathbb{R}^{512}$，依靠 Transformer 的自注意力隐式建模时间导数。

该架构由以下部分组成：

输入投影：含噪轨迹 $\mathbf{v}t$ 经线性投影至 $d_{\text{model}} = 256$ 维，并与可学习的位置编码相结合：

$$\mathbf{h}_0 = \text{Linear}(\mathbf{v}_t) + \text{PE}$$

时间嵌入：扩散时间步 $t$ 通过正弦编码（与 U-Net 模型中的方法相同）进行嵌入，并广播相加至所有序列位置：

$$\mathbf{h}_0 := \mathbf{h}_0 + \text{TimeEmbed}(t)$$

条件注入：条件向量$\mathbf{c} = [\bar{v}/30, T/1000, v_{max}/40, d_{veh}] \in \mathbb{R}^4$（经过归一化）被嵌入至 256 维，并通过交叉注意力注入每个 Transformer 层。

*Transformer* 编码器：包含六层多头自注意力（8 个头）和前馈网络（$d_{ff}$ =1024）：

$$\begin{aligned}
\mathbf{q} &= \text{Heads}(\text{SelfAttn}(\mathbf{h}_\ell)) \\
\mathbf{h}_{\ell+1} &= \text{LayerNorm}(\mathbf{h}_\ell + \mathbf{q}) \\
\mathbf{h}_{\ell+1} &= \text{LayerNorm}(\mathbf{h}_{\ell+1} + \text{FFN}(\mathbf{h}_{\ell+1}))
\end{aligned}$$

输出投影：最终的隐藏状态经过投影后用于预测噪声 $\epsilon_\theta \in \mathbb{R}^{512}$。

该模型约有 550 万个参数，比 U-Net（~8M）更小，但由于采用全局注意力，具有更大的感受野。

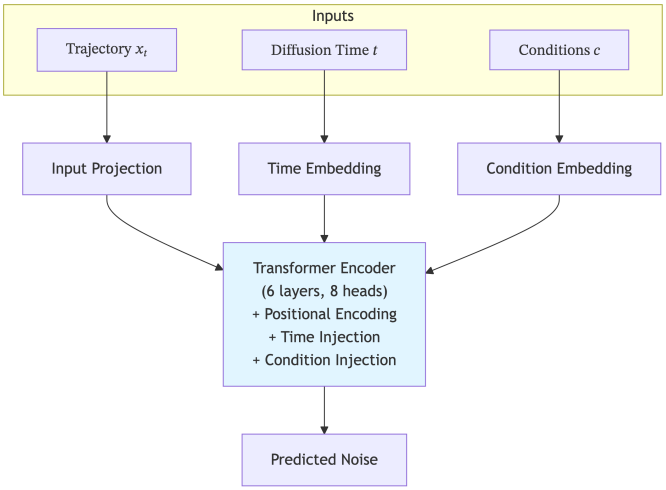



图5：CSDI Transformer 架构（550 万个参数）。含噪轨迹 $x_t$、扩散时间 $t$ 和条件 $c$ 经过嵌入后，通过 6 个带位置编码、时间注入和条件注入的 Transformer 编码器层进行处理。输出为预测噪声 $\hat{\epsilon}$。”

反向扩散（采样）：生成过程通过迭代去噪，从纯噪声逐步生成干净轨迹：

$$\mathbf{x}_T \sim \mathcal{N}(0, \mathbf{I}) \to \mathbf{x}_{T-1} \to \cdots \to \mathbf{x}_1 \to \mathbf{x}_0 \text{ (clean trajectory)}$$

与 U-Net 模型类似，我们通过修复在每个反向步骤施加硬边界约束，确保在整个去噪轨迹中满足$v_0 = v_T = 0$ 。

*1*）物理信息训练：不同于失败的 PID 扩散方法，CSDI 通过精心加权的辅助损失成功整合了物理约束。总训练目标为：

$$\begin{aligned}
\mathcal{L}_{\text{CSDI}} = \mathcal{L}_{\text{MSE}} &+ 0.1\mathcal{L}_{\text{smooth}} + 0.03\mathcal{L}_{\text{accel}} \\
&+ 0.02\mathcal{L}_{\text{jerk}} + 0.05\mathcal{L}_{\text{accel_dist}}
\end{aligned}$$

其中：

去噪损失：

$$\mathcal{L}_{\text{MSE}} = \|\epsilon - \epsilon_\theta(\mathbf{v}_t, t, \mathbf{c})\|^2$$

时间平滑损失（二阶导数惩罚）：

$$\mathcal{L}_{\text{smooth}} = \sum_t (v_{t+2} - 2v_{t+1} + v_t)^2$$

加速度惩罚 （对极端值的软约束）：

$$\mathcal{L}_{\text{accel}} = \sum_t [\text{ReLU}(a_t - 4.0)^2 + \text{ReLU}(-a_t - 5.0)^2]$$

其中 $a_t = v_{t+1} - v_t$。正加速度上限为 4 m/s2，制动上限为 5 m/s2。

加加速度惩罚（舒适性和真实性）：

$$\mathcal{L}_{\text{jerk}} = \sum_t \text{ReLU}(|j_t| - 2.0)^2, \quad j_t = a_{t+1} - a_t$$

加速度分布匹配：

$$\mathcal{L}_{\text{accel_dist}} = (\sigma_a^{\text{pred}} - \sigma_a^{\text{target}})^2$$

其中，$\sigma_a^{\text{target}} \approx 0.5$ m/s2 为经验标准差。

CSDI 能够成功，而硬约束扩散模型会失败，原因有三：(1) 阈值激活惩罚（ReLU门控）允许在物理边界内自然变化，而不是强行施加僵硬的等式约束；(2) 与局部卷积相比，Transformer 的全局自注意力更能适应长程运动学一致性；(3) 通过仔细调整权重，确保物理损失比主要的均方误差目标小 20-200×，从而避免梯度主导。在 A100 上训练 200 个周期约需 12 小时。

车辆类型兼容性：物理约束（4/5 m/s2）的设定旨在涵盖所有常见车辆类型的加速能力：

表 III：车辆类型兼容性与加速度限制

| 车辆 | 最大加速度 | 最大减速度 |
|---|---|---|
| 跑车 | 4-6 m/s$^2$ | 10-12 m/s$^2$ |
| 乘用车 | 2.5-4 米/秒$^2$ | 8-10 米/秒$^2$ |
| SUV | 2-3.5 米/秒$^2$ | 7-9 米/秒$^2$ |
| Bus | 1-2 米/秒$^2$ | 4-6 米/秒$^2$ |
| 重型卡车 | 0.5-1.5 m/s$^2$ | 3-5 m/s$^2$ |

这些数值与公认的车辆动力学特性（Gillespie 1992）一致，确保模型生成的轨迹在 CMAP 数据集中遇到的各类道路车辆范围内均符合物理规律。

加权条件采样（增强）：为了解决训练数据中某些速度/持续时间范围代表不足的数据不平衡问题，我们在生成过程中采用重要性采样。条件的采样概率$c$ 按以下方式加权：$P(c) \propto ($平均速度$)^{\beta_s} \cdot ($持续时间$)^{\beta_d}$，其中增强参数 $\beta_s$、$\beta_d \in [0, 2]$ 用于控制对高速或长持续时间行程的过采样。对于 U-Net 扩散模型，使用 $\beta_s = 1.75$ 后，我们有效填补了速度直方图中稀疏的“蓝色缺口”（25-30 m/s）。CSDI 模型采用均匀采样（$\beta = 1.0$），因为其较大的训练集已能提供充分覆盖。

*2*）运动学质量优化：CSDI 模型的开发针对三个质量问题进行了改进：(1) 标准插补目标函数导致的高频抖动，通过时间平滑损失$\mathcal{L}_{\text{smooth}}$（权重为 0.1）、增加扩散步数（100→200）以及后处理高斯平滑予以解决；(2) 过度均匀平滑导致的边界和峰值失真，则通过减小核大小并采用条件边界斜坡予以解决——仅当端点并非已接近零时才应用；(3) 加速度分布不匹配，通过阈值触发的物理惩罚项（$\mathcal{L}_{\text{accel}}$、$\mathcal{L}_{\text{jerk}}$、$\mathcal{L}_{\text{accel_dist}}$）、扩展包含车辆动力学的四维条件输入、提高模型容量（4→6 个 Transformer 层，$d_{ff}$ 512→1024）以及针对重型车辆的数据增强予以解决。余弦噪声调度取代了线性调度。这些改进使 WD 速度达到 $= 0.30$，并确保了严格的物理有效性。

*3) CSDI* 采样与后处理：CSDI 使用与 U-Net 模型相同的 DDPM 反向扩散过程，但采用余弦噪声调度（Nichol 和 Dhariwal，2021）：

$$\bar{\alpha}_t = \frac{f(t)}{f(0)}, \quad f(t) = \cos^2\left(\frac{t/T_{diff} + s}{1+s} \cdot \frac{\pi}{2}\right)$$

其中，$s = 0.008$ 是用于防止奇点的小偏移量。余弦调度在高信噪比区域集中安排更多扩散步数，从而提高采样质量。

后处理包括四个步骤：1. 高斯平滑：使用 $\sigma = 1.5$ 的一维高斯滤波器，核大小为 7，在保留加速度峰值的同时去除高频采样伪影。2. 条件边界斜坡：如果 $v_0 > 0.5$ 或 $v_T > 0.5$，则在 3 秒内应用线性斜坡，以强制端点为零。如果端点已经接近零，则跳过此步骤，以避免过度平滑。3. 车辆感知平滑：对于重型车辆（$d_{veh} < 0.4$），应用额外的平滑处理（核 =9），以反映较慢的动力学。4. 相关噪声（可选）：为确保较大的加速度尾部与真实数据相匹配，我们允许在扩散采样后添加时间相关高斯噪声。该噪声通过对具有 10 秒相关长度的白噪声进行平滑，并将其缩放至较小幅度（$\sigma_{corr} \approx 0.03$）而生成，从而防止分布变得“过于保守”或过于狭窄，同时保持时间一致性。在我们的最终模型中，物理训练降低了对该步骤的需求（默认为 $\sigma_{corr} = 0.0$），但它仍是微调方差的有效调节手段。

使用 CSDI 在 A100 上生成 1000 条轨迹约需 3 分钟；由于模型参数更少且 Transformer 推理效率更高，速度快于 U-Net 扩散模型。

### *D.* 替代生成方法

我们还评估了 DoppelGANger（Lin 等人，2020）、SDV 的 PARSynthesizer（Patki、Wedge 和 Veeramachaneni，2016）以及 Chronos（Ansari 等，2025年）作为基线方法。DoppelGANger 出现了模式崩溃（87% 的轨迹属于高速公路行驶状态）；SDV 则受到曝光偏差影响，导致时间不连续性；Chronos 缺乏用于轨迹合成的条件机制。所有基线方法的定量结果见表 IV。

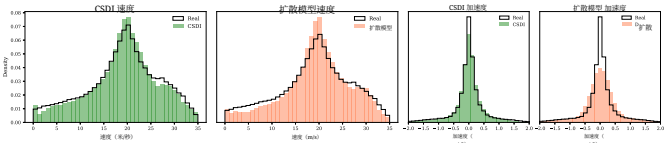
## V. 结果

我们将 6,367 个微行程按簇分层，划分为 80% 的训练集（5,094 个行程）和 20% 的测试集（1,273 个行程）

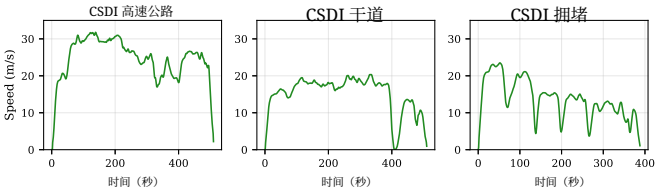
以确保驾驶工况的代表性均衡。所有模型均在训练集上训练，并通过生成 1,273 条以测试集实际 $(\bar{v}, T)$ 值为条件的合成轨迹进行评估。该方案确保了公平比较：每个模型都仅根据聚合条件信息，尝试重现测试分布。

所有训练和轨迹生成实验均在配备 40GB 显存的 NVIDIA A100 GPU 上进行，使用 CUDA 12.1、PyTorch 2.0 和混合精度（FP16）计算。马尔可夫基线仅使用 CPU 上的 NumPy 运算实现。

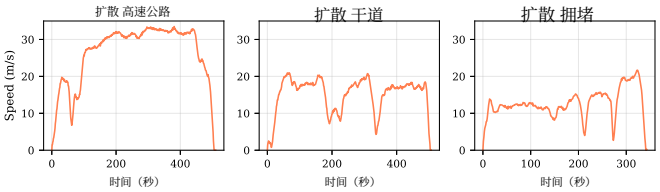
评估采用涵盖三个维度的综合框架。分布保真度通过速度、加速度和车辆比功率（VSP）分布的瓦瑟斯坦距离（WD）、速度-加速度频率分布（SAFD）的二维 WD、采用 RBF 核（带宽为 1.0）的最大均值差异（MMD），以及 VSP 的柯尔莫哥洛夫-斯米尔诺夫统计量进行评估。VSP 是一种车辆功率需求指标（单位为 kW/吨），考虑了空气阻力、滚动阻力和道路坡度，常用于排放和能耗评估。运动学有效性通过边界违规率（行程起点或终点速度未处于零值 ±0.1 m/s 范围内的行程百分比）、用于量化平滑度的对数无量纲加加速度（LDLJ）（数值越低表示轨迹越平滑）、最大速度和加速度标准差进行衡量。真实性和效用则通过判别分数（使用随机森林分类器区分真实数据与合成数据；分数为 0.5 表示两者完全不可区分）以及合成训练真实测试（TSTR）平均绝对误差（MAE）进行评估。TSTR 中，预测模型在合成数据上训练，并在真实数据上测试，以衡量其对下游任务的效用。



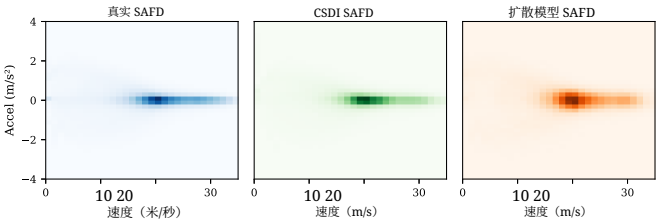
(a) 速度/加速度分布（CSDI 与扩散模型）




(b) CSDI 轨迹（高速公路、干道、拥堵）




(c) 扩散模型轨迹（高速公路、干道、拥堵）




(d) SAFD 热力图（真实数据、CSDI、扩散模型）

图 6：主要结果的综合比较。第 *1* 行：(a-b) 速度和加速度分布，展示了 CSDI 更优的匹配效果。第 *2* 行：(c-e) CSDI 在不同工况下的样本轨迹。第 *3* 行：(f-h) 扩散模型的样本轨迹。第 *4* 行：(i-k) 速度-加速度频率分布（SAFD），展示真实数据、CSDI 和扩散模型的联合分布。

### *A.* 主要结果

表IV 展示了所有模型和指标的全面比较。

粗体 表示性能最佳（最接近真实值或目标值）。

CSDI 的整体分布匹配效果最佳，WD 速度为 0.30，几乎比 U-Net 扩散模型的 0.5622 好一倍，也显著优于马尔可夫基线。加速度分布匹配效果尤为出色（WD 加速度 $= 0.026$），体现了经过精心设计的物理信息训练。U-Net 扩散模型的表现也很好，尤其在二维 SAFD 指标（0.0005）上表现出色，

表明准确捕捉了联合速度-加速度流形。

马尔可夫基线取得了尚可的 WD 速度（1.82），但在加速度方面表现不佳（0.145），这证实离散化伪影和无记忆转移无法捕捉运动学平滑性。Chronos 尽管其基础模型预训练，表现仍不如马尔可夫（WD 速度 2.15），这表明未经领域特定微调的零样本迁移是不够的。DoppelGANger 和 SDV 表现极差（WD 速度 $> 2.8$），DoppelGANger 的模式崩溃产生了不切实际的高速公路占比较高的分布。

所有具备物理感知能力的模型（U-Net 扩散模型、CSDI、马尔可夫模型）都通过各自的约束机制（修复、后处理、桥采样）完美满足边界条件。相比之下，自回归模型（Chronos 23.4%、SDV 23.0%）和对抗方法（DoppelGANger 12.1%）难以满足硬约束。

CSDI 生成的轨迹最平滑（LDLJ $=$ - 3.85，最接近真实值 - 3.92），这验证了其时间平滑性和加加速度惩罚的有效性。U-Net

表 IV：生成模型的比较性能

| 指标 | Real | U-Net 扩散模型 | CSDI | 马尔可夫链 | Chronos | DoppelGANger | SDV |
|---|---|---|---|---|---|---|---|
| **分布保真度** | | | | | | | |
| WD 速度 | - | **0.5622** | **0.30** | 1.82 | 2.15 | 3.42 | 2.87 |
| WD 加速度 | - | **0.0800** | **0.026** | 0.145 | 0.198 | 0.312 | 0.421 |
| WD 垂直速度 | - | **1.5517** | 1.89 | 2.34 | 3.12 | 4.87 | 3.65 |
| WD 二维 SAFD | - | **0.0005** | 0.0008 | 0.0023 | - | - | - |
| MMD ($\times 10^{-3}$) | - | **<0.1** | **0.12** | 2.34 | 3.87 | 8.92 | 6.15 |
| KS 垂直速度 | - | **0.0631** | 0.072 | 0.123 | 0.187 | 0.298 | 0.245 |
| **运动学有效性** | | | | | | | |
| 边界违规 | 0% | 0% | 0% | 0% | 23.4% | 12.1% | 23.0% |
| LDLJ（平滑度） | -3.92 | **-3.50** | -3.85 | -0.15 | -2.12 | -4.10 | -1.15 |
| 最大速度 (m/s) | 31.6 | 32.1 | 31.8 | 34.5 | 32.5 | 31.9 | 45.2 |
| 加速度标准差 (m/s²) | 0.51 | **0.53** | **0.49** | 2.10 | 0.85 | 0.35 | 4.12 |
| **效用** | | | | | | | |
| 判别分数（0.5 为理想值） | 0.50 | 0.62 | **0.49** | 0.85 | 0.78 | 0.51 | 0.99 |
| TSTR 平均绝对误差（km/h） | - | 4.2 | **2.1** | 8.5 | 5.4 | 12.3 | 15.2 |

扩散模型的 LDLJ（-3.50）表明其平滑度相近，且加速度方差仍然准确（0.53，真实值为 0.51）。由于离散化，马尔可夫基线生成的轨迹平滑度较低（LDLJ = -0.15），而生成对抗网络/自回归方法则表现出不同的平滑度特征。

表中的最大速度显示，所有物理感知模型（U-Net 扩散模型为 32.1 m/s，CSDI 为 31.8 m/s）都接近真实数据的最大值（31.6 m/s）。SDV 生成的速度过高（45.2 m/s），表明其约束执行不足。

在深度模型中，CSDI 的训练效率最高（12 小时，而 U-Net 扩散模型为 21.5 小时），这得益于 Transformer 的全局感受野，因此所需训练轮数较少。其生成速度也更快（3 分钟，而 U-Net 扩散模型为 5 分钟），这对大规模仿真至关重要。马尔可夫基线的计算效率仍然最高，适用于可解释性比分布保真度更重要的场景。Chronos 虽然无需训练，但推理速度较慢，因为这个拥有 6000 万参数的大型模型采用自回归方式生成。

判别分数用于量化区分合成轨迹与真实轨迹的难度。CSDI 的得分为 0.49，表现出很高的不可区分性（目标值为 0.5）。U-Net 扩散模型的得分为 0.62，表明合成样本较容易被识别，但仍然非常逼真。马尔可夫基线（0.85）生成的轨迹具有更多可检测伪影，而 DoppelGANger（0.51）尽管存在其他质量问题，仍实现了良好的不可区分性。

TSTR（合成训练真实测试）用于衡量下游效用：在合成数据上训练的模型能否在真实数据上表现良好？CSDI 的平均绝对误差最低（2.1 km/h），U-Net 扩散模型紧随其后（4.2km/h），这表明合成轨迹保留了预测建模所需的统计特性。这验证了它们在能耗评估和仿真任务中的应用价值。

中。

我们通过在固定持续时间的情况下，生成目标平均速度各不相同的轨迹，评估了条件控制能力。图 6 显示，CSDI 和扩散模型均能在 10–25 m/s 的范围内准确匹配目标速度（平均绝对误差 $< 0.5$ m/s）。CSDI 的车辆动力学条件控制还支持更进一步的控制：使用$d_{veh} = 0.2$ （重型卡车）而非 $d_{veh} = 0.8$ （跑车）进行生成，会产生不同的加速度曲线；卡车轨迹的加速度较低（均值为 1.2 m/s2，而跑车为 3.1 m/s2），动力学也更平滑（LDLJ 分别为 -12.3 和 -10.1）。

最后，表 V 比较了训练和生成所需的计算资源。

## VI. 讨论

本研究表明，扩散模型——尤其是基于 Transformer 的 CSDI 架构——适用于需要合成驾驶数据的智能交通系统应用。通过与传统基线方法以及 DoppelGANger 和 SDV 等现代方法进行系统比较，我们证实，采用物理信息训练的 CSDI 表现出色：瓦瑟斯坦距离指标比基线方法低 2–6× ，能够很好地满足边界条件，并准确匹配真实驾驶动力学的平滑度。CSDI 的优势源于其联合生成完整序列的能力，从而避免自回归误差累积；同时，它利用 Transformer 自注意力捕捉长距离运动学依赖关系。

此外，我们的研究总结了对从业者具有重要意义的经验教训。为避免优化冲突，应通过软性的阈值激活惩罚来整合物理约束，而非采用硬性约束；与局部卷积相比，Transformer 架构的全局感受野更适合处理长距离运动学依赖关系。我们还观察到，对抗性训练

表 V：计算效率

| 模型 | 参数 | 训练时间 | 生成时间 | GPU 内存 | 硬件 |
|---|---|---|---|---|---|
| **U-Net 扩散模型** | 8.2M | 21.5 小时 | 约 5 分钟 | 12 GB | A100 |
| **CSDI** | 5.5M | 12 小时 | 约 3 分钟 | 8 GB | A100 |
| **马尔可夫** | - | <5分钟（CPU） | <1分钟（CPU） | - | CPU |
| **Chronos** | 6000 万（冻结） | 0（预训练） | 约 10 分钟 | 16 GB | A100 |
| **DoppelGANger** | 12M | 8 小时 | 约 8 分钟 | 10 GB | A100 |
| **SDV** | 15M | 4 小时 | 约 2 分钟 | 6 GB | A100 |

和自回归分解在架构上并不适合高保真、物理约束合成，往往会导致模式崩溃或时间不连续性。除了提供经过验证的能耗评估和交通仿真工具外，公开发布的实现代码和训练好的模型还确立了基于扩散模型的合成方法，是适用于智能交通系统应用的稳健框架。

本研究受到若干限制。512 秒的轨迹长度上限排除了较长的行程，因此高速公路通勤需要采用滑动窗口生成。模型主要基于乘用车训练，仍需使用卡车和公共汽车数据集进行验证。道路类型条件是隐式的，而非显式的。最后，尽管 2007 年 CMAP 数据集年代较久，但模型捕捉到了微观运动学特征——加速能力和跟车动力学；这些特征受不变的车辆物理特性支配，而非不断变化的交通模式。TSTR 结果证实，这些物理原理能够迁移至下游任务。

若干扩展可增强模型的能力和效率。首先，采用去噪扩散隐式模型（DDIM）（J. Song、Meng 和 Ermon，2021 年）等更快的采样技术，可将生成步骤从 1000 步减少到 50–100 步，实现实时应用所需的 10–20× 倍加速。其次，引入多模态道路条件（例如高速公路、干道和本地道路），可对驾驶模式进行显式控制，从而支持生成逼真的多路段行程。第三，将框架扩展至包含交互建模的多智能体场景，将有助于交通仿真应用；这一方向可借鉴扩散模型在自动驾驶领域的最新进展（Feng 等，2023年），以及面向智能交通系统网络的多尺度生成式 Transformer（Adam 等，2025年）。第四，在车辆轨迹数据上微调 Chronos 等基础模型，并针对边界约束进行调整，可利用大规模预训练来缩短训练时间。最后，将道路高程和坡度曲线作为额外条件纳入模型，将显著提高能耗建模的准确性，因为坡度是车辆功率需求的主要影响因素。总体而言，这些方向将推动综合性合成交通生成技术的发展，以支持智能交通系统评估与设计的各类任务。

## VII. 数据与代码可用性

所有代码、训练好的模型权重（U-Net 扩散模型、CSDI）和评估脚本均可在 https://github.com/VadimSokolov/diffusion-trajectory-generation 获取。CMAP2007 数据集可通过 NREL 的交通安全数据中心（TSDC）公开获取。


Adam, Abuzar B. M., Tahir Kamal, Mohammed A. M. Elhassan, Abdullah Alshahrani, Saeed Hamood Al-samhi, and Ahmed Aziz. 2025. “面向 AAV 辅助智能交通网络的多尺度生成式 Transformer 原始-对偶 PPO 框架。”*IEEE* 智能交通系统汇刊, 1–16.

Alcaraz, Juan Miguel Lopez 和 Nils Strodthoff。2022 年。《基于扩散的时间序列插补与结构化状态空间模型预测》。*arXiv*预印本 *arXiv:2208.09399*。https://arxiv.org/abs/2208.09399。Altché, Florent 和 Arnaud de La Fortelle。2017年。《用于高速公路轨迹预测的长短期记忆网络》。载于*2017*年*IEEE*第*20*届智能交通系统国际会议（*ITSC*），353–59。IEEE。Ansari, Abdul Fatir、Oleksandr Shchur、Jaris Küken、Andreas Auer、Boran Han、Pedro Mercado、Syama Sundar Rangapuram 等。2025年。《Chronos-2：从单变量预测到通用预测》。arXiv。https://arxiv.org/abs/2510.15821。Auld, Joshua、Michael Hope、Hubert Ley、Vadim Sokolov、Bo Xu 和 Kuilin Zhang。2016年。《POLARIS：用于出行需求、交通网络及运行仿真的集成式基于智能体建模框架的开发与实现》。交通研究*C*辑：新兴技术64：101–16。Auld, Joshua、Michael Hope、Hubert Ley、Bo Xu、Kuilin Zhang 和 Vadim Sokolov。2013年。《区域综合交通网络与基于活动的需求仿真建模框架（Polaris）》。载于下一代基础设施国际研讨会。Auld, Joshua、Dominik Karbowski、Vadim Sokolov 和 Namwook Kim。2016年。《用于评估区域层面交通能源影响的非集计模型系统》。载于交通研究委员会第*95*届年会。

Behnia, Farnaz, Dominik Karbowski 和 Vadim Sokolov。2023年。“用于车辆速度轨迹生成的深度生成模型——

迹。”《商业与工业中的应用随机模型》39 (5): 701–19。Chen, Jun, Michal Weiszer, Paul Stewart 和 Masiha-lah Shabani。2016。“通过主动路径规划实现更真实、更具成本效益且更环保的地面运行——第一部分：最优速度曲线生成。”《*IEEE*智能交通系统汇刊》17 ( 5): 1196–1209。芝加哥大都会规划署。2008年。“ CMAP区域家庭出行清单。”交通安全数据中心，美国国家可再生能源实验室。Dauner, Daniel, Marcel Hallgarten, Andreas Geiger 和 Kashyap Chitta。2023。“摒弃对基于学习的车辆运动规划的误解。” arXiv。https://arxiv.org/abs/2306.07962。Durham, Garland B, 和 A Ronald Gallant。2002。“连续时间扩散过程最大似然估计的数值技术。”《商业与经济统计学杂志》20 (3): 297–338。Feng, Lan, Quanyi Li, Zhenghao Peng, Shuhan Tan 和 Bolei Zhou。2023。“TrafficGen：学习生成多样且逼真的交通场景。”载于*2023*年*IEEE*机器人与自动化国际会议（*ICRA*），3567–75。IEEE。Gillespie, Thomas D. 1992年。《车辆动力学基础》。宾夕法尼亚州沃伦代尔：汽车工程师协会。Ho, Jonathan, Ajay Jain 和 Pieter Abbeel。2020。“去噪扩散概率模型。”载于《神经信息处理系统进展》，33:6840–51。Ho, Jonathan 和 Tim Salimans。2022。“无分类器扩散引导。” *arXiv*预印本 *arXiv:2207.12598*。https://arxiv.org/abs/2207.12598。Huang, Xianan, Boqi Li, Huei Peng, Joshua A Auld 和 Vadim O Sokolov。2020。“通过拼车实现生态按需出行车队控制。”《*IEEE* 智能交通系统汇刊》 23 (4): 3158–68。Huang, Yanjun, Jian Chen, Chang Huang, Xinggang Wang, Wenyu Liu 和 Jianqiang Huang。2022。“自动驾驶轨迹预测方法综述。”《*IEEE*智能交通系统汇刊》。Karbowski, Dominik, Namwook Kim, Joshua Auld 和 Vadim Sokolov。2016。“评估交通管理和车辆混合动力化对能耗的影响。”《应用科学与技术中的复杂性国际期刊》1 (1): 107–24。Karbowski, Dominik, Aymeric Rousseau, Vivien Smis-Michel 和 Valentin Vermeulen。无日期。“利用地理信息系统进行车辆能源效率行程预测。”载于 第*21*届世界智能交通系统大会（底特律，密歇根州，*2014*年*9*月*7*日—*2014*年*9*月*11*日），–,. Karbowski, Dominik, Vadim Sokolov 和 Jeong Jongryeol。2016。“插电式混合动力汽车基于最优路径的控制的节油潜力。” *IFAC-PapersOnLine* 49（11）：128–33。Karbowski, Dominik、Vadim Sokolov和Aymeric Rousseau。2015。“通过数字地图和连接性实现车辆能量管理优化。”阿贡国家实验室（ANL），伊利诺伊州阿贡（美国）。Lan, Wenxing、Jialin Liu、Bo Yuan和Xin Yao。2025。“面向自动驾驶的可控多模态运动行为生成。” *IEEE*智能交通系统汇刊，1–16。Li, Quanyi、Zhenghao Peng、Lan Feng、Zhizheng Liu、Chenda Duan、Wenjie Mo和Bolei Zhou。2023。“ScenarioNet：面向大规模交通场景仿真与建模的开源平台。”arXiv。https://arxiv.org/abs/2306.12241。Liao, Bencheng、Shaoyu Chen、Xinggang Wang、Tianheng Cheng、Qian Zhang、Wenyu Liu和Chang Huang。2024。“DiffusionDrive：面向端到端自动驾驶的截断扩散模型。”*arXiv*预印本 *arXiv:2406.07806*。https://arxiv.org/abs/2406.07806。Lin, Zinan、Alankar Jain、Chen Wang、Giulia Fanti和Vyas Sekar。2020。“使用生成对抗网络共享网络化时间序列数据：挑战、初步前景与开放问题。”载于*ACM*互联网测量会议（*IMC*）论文集，464–83。ACM。Moawad, Ayman、Zhijian Li、Ines Pancorbo、Krishna Murthy Gurumurthy、Vincent Freyermuth、Ehsan Islam、Ram Vijayagopal、Monique Stinson和Aymeric Rousseau。2021。“一种实时节能降本的车辆路径分配神经推荐系统。”arXiv。https://arxiv.org/abs/2110.10887。Mozaffari, Sajjad、Omar Y Al-Jarrah、Alexandros Mouzakitis、Phil Jennings和Stratis Kanarachos。2020。“基于深度学习的车辆行为预测在自动驾驶应用中的综述。”*IEEE*智能交通系统汇刊。Nichol, Alexander Quinn和Prafulla Dhariwal。2021。“改进的去噪扩散概率模型。”载于国际机器学习会议，8162–71。PMLR。Papamakarios, George、Eric Nalisnick、Danilo Jimenez Rezende、Shakir Mohamed和Balaji Lakshminarayanan。2019。“用于概率建模与推理的归一化流。”*arXiv:1912.02762 [Cs、Stat]*,12月。https://arxiv.org/abs/1912.02762。Patki, Neha、Roy Wedge和Kalyan Veeramachaneni。2016。“Synthetic Data Vault。”载于*2016*年*IEEE*数据科学与高级分析国际会议（*DSAA*），399–410。IEEE。Perez, Ethan、Florian Strub、Harm De Vries、Vincent Dumoulin和Aaron Courville。2018。“FiLM：基于通用条件层的视觉推理。”载于*AAAI*人工智能会议论文集。第32卷。1。

Qian, Yu、Xunhao Li、Jian ... Zhang和Maoze Wang。2025。“一种用于时空速度插补与轨迹重建的Diffusion-TGAN框架。”*IEEE*智能交通系统汇刊 26（11）：18948–62。Rasul, Kashif、Calvin Seward、Ingmar Schuster和Roland Vollgraf。2021。“用于多元概率时间序列预测的自回归去噪扩散模型。”载于国际机器学习会议（*ICML*），8857–68。PMLR。Rong, Yi、Yingchi Mao、Yinqiu ... Liu和Dusit Niyato。2025。“ICST-DNET：一种用于交通速度预测的可解释因果时空扩散网络。”*IEEE*智能交通系统汇刊 26（7）：9781–98。Schultz, Laura和VadimSokolov。2018。“面向动态城市交通问题的深度强化学习。”*arXiv*预印本 *arXiv:1806.05310*。https://arxiv.org/abs/1806.05310。Sokolov, Vadim、Joshua Auld和Michael Hope。2012。“一种采用基于智能体方法开发综合交通系统模型的灵活框架。”*Procedia Computer Science* 10：854–59。Song, Jiaming、Chenlin Meng和Stefano Ermon。2021。“去噪扩散隐式模型。”载于国际学习表征会议。Song, Yang、Jascha Sohl-Dickstein、Diederik P. Kingma、Abhishek Kumar、Stefano Ermon和Ben Poole。2020。“通过随机微分方程进行基于分数的生成建模。”*arXiv*预印本 *arXiv:2011.13456*。https://arxiv.org/abs/2011.13456。Suo, Simon、Sebastian Regalado、Sergio Casas和Raquel Urtasun。2021。“TrafficSim：学习仿真逼真的多智能体行为。”arXiv。https://arxiv.org/abs/2101.06557。Tashiro, Yusuke、Jiaming Song、Yang Song和Stefano Ermon。2021。“CSDI：用于概率时间序列补全的条件基于分数的扩散模型。”载于《神经信息处理系统进展》，34：24804–16。Wu, Jian、Carol Flannagan、Ulrich Sander和Jonas Bärgman。2025。“利用碰撞前数据基于模型生成覆盖全严重程度范围的代表性追尾碰撞场景。”*IEEE*智能交通系统汇刊 26（10）：15932–50。Yoon, Jinsung、Daniel Jarrett和Mihaela van der Schaar。2019。“时间序列生成对抗网络。”载于《神经信息处理系统进展》*32*，由H. Wallach、H. Larochelle、A. Beygelzimer、F. d\textquotesingle Alché-Buc、E. Fox和R. Garnett编辑，5508–18。Curran Associates, Inc.。Zhao, Tianyang、Yifei Xu、Mathew Monfort、Wongun Choi、Chris Baker、Yibiao Zhao、Yizhou Wang和Ying Nian Wu。2019。“面向情境轨迹预测的多智能体张量融合。”载于论文集*IEEE/CVF* 计算机视觉与模式识别会议（*CVPR*），5559–68。