arXiv:2301.02561v1 [cs.RO] 6 Jan 2023

利用状态与意图信息进行交叉路口多车辆轨迹预测

Dekai Zhu[1]*、Qadeer Khan[1]* 和 Daniel Cremers[1]

摘要—传统的道路交通参与者未来轨迹预测方法依赖于已知其过去轨迹的信息。相比之下，本研究仅利用当前状态和行驶意图来预测交叉路口处多辆车辆的轨迹。此外，在车辆之间传递这些信息，可使每辆车更全面地了解环境，从而做出更准确的预测。为此，我们训练了一种神经网络，利用多辆车辆的状态和意图来预测其未来轨迹。将意图作为输入，还使我们的方法能够进一步控制多辆车辆驶向期望路径。实验结果表明，我们的方法在交叉路口的轨迹预测和车辆控制方面均具有鲁棒性。本研究完整的训练和评估代码可在此处获取：https://github.com/Dekai21/Multi_智能体_交叉路口。

索引词—轨迹预测，多车辆，神经网络，深度学习

I. 引言

过去十年间，深度学习朝着实现完全自动驾驶这一终极目标取得了长足进步 [1]。自动驾驶车辆配备了多种不同的传感器，例如雷达、GPS、惯性测量单元、激光雷达、摄像头，或它们的组合，用于目标检测、分类、定位和导航等任务 [2], [3], [4], [5],[6]。其中，基于视觉的传感器（摄像头、激光雷达等）已被证明最有望实现与人类驾驶水平相当的表现。这是因为，在感知驾驶环境方面，这些传感器最接近模拟人类视觉的特性。结合其他传感器，它们已成功应用于紧急制动 [7], [8], 车道保持 [9], [10],行人检测、目标跟踪 [11], [12]等多种任务。然而，安装在自车上的此类视线传感器主要用于涉及单辆车的任务，因此自身存在若干局限：

- 由于视场受限、存在遮挡等原因，它们只能部分观测环境。因此，在交通交叉口等繁忙区域执行机动操作可能并不可行。这一点很重要，因为相当一部分车辆碰撞发生在交通交叉口 [13]，而且这类碰撞往往也更为严重[14]。
- 每辆车辆都配有独立的传感器和单独的处理设备。因此，所有车辆所需的总计算能力会很高。

[1] 慕尼黑工业大学 CIT 计算机视觉组。本研究由慕尼黑机器学习中心资助。* 这些作者贡献相同

此外，这些资源会占用自车内部空间，甚至可能需要冷却。

- 仿真引擎在测试和评估自动驾驶算法方面发挥了关键作用。然而，仿真中渲染的图像等传感器数据可能无法真实反映现实。因此，这种领域偏移会妨碍相关技术在现实世界中的部署。

为了解决交叉路口等关键区域的部分可观测性问题，可以采用鸟瞰视图（BEV）的方式部署摄像头，同时观测场景中的所有智能体，如图 1 所示。此类俯视图像常用于轨迹预测 [15], [16]。通过永久安装在高处基础设施上的摄像头 [17], [18] ，或使用无人机影像，可以获取智能体的状态，精度可达厘米级 [19]，并利用基于视觉的目标检测算法获取这些状态信息。随后，可利用每辆车的状态信息预测其未来轨迹或控制动作序列。需要注意的是，每辆车也可以在采取适当行动之前汇总其他智能体的信息。与使用 RGB 摄像机等自车搭载传感器相比，使用准确的状态信息还有两项优势：1）可以减轻资源的计算负担，因为不再需要由每辆车独立处理包含大量像素的图像。2）由仿真图像渲染结果与现实不符所导致的域偏移问题不再构成顾虑。这是因为我们使用车辆的状态（位置、朝向等）作为抽象表示来承载相关信息。因此，借助这种抽象，我们便能够在一个领域上训练模型，并在另一个领域上进行测试，正如我们在实验中所展示的那样。

图1 进一步说明，所有车辆的状态信息以及它们直行、左转或右转的预期意图都会传递给多车辆轨迹预测（MTP）模块。MTP 模块根据所提供的信息预测每辆车辆的未来轨迹。在 MTP 内部，未来轨迹预测由聚合模块完成，该模块在所有车辆之间共享权重。因此，模型能够处理场景中任意数量的车辆。请注意，要预测某一特定车辆的轨迹，聚合模块不仅会获取该车辆的信息，还会通过消息传递考虑其他车辆的信息 [20]。这使每辆车辆都能全面了解

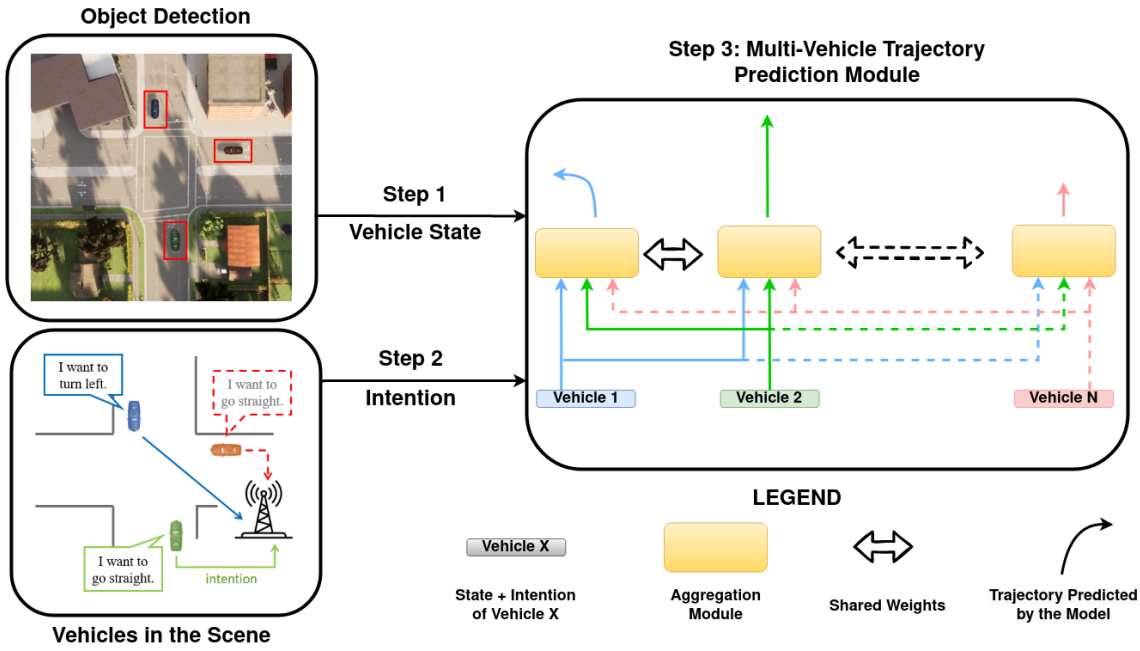


图1：**多车轨迹预测框架：** 步骤*1*：在交叉路口的俯视图像上进行目标检测，以提取场景中每辆车的状态信息。这些信息会发送至多车轨迹预测（MTP）模块。步骤*2*：与此同时，场景中的每辆车也会将其意图发送至MTP。意图信息会告知MTP，某辆车是否打算在即将到来的交叉路口左转、右转或直行。步骤*3*：随后，MTP会将状态与意图信息的组合传递给聚合子模块，以预测每辆车的未来轨迹。请注意，每辆车的轨迹预测不仅取决于其自身的状态与意图信息，也会考虑其他车辆的相关信息。本文聚焦于MTP模块，并展示如何利用多辆车的状态与意图信息，同时预测它们的未来轨迹。

环境，从而做出有依据的轨迹预测。相比之下，图2 展示了在进行轨迹预测时不聚合其他车辆信息所产生的影响。为此，本文的贡献总结如下：

1）我们证明，仅使用状态与意图信息的方法优于使用过去轨迹信息的方法。2）我们的模型能够预测任意数量车辆的未来轨迹。它会聚合来自其他车辆的信息，从而提供更准确的预测。3）我们展示了该模型可以在一个平台上训练，并在另一个平台上测试。4）我们的未来轨迹预测方法可以轻松扩展，以便在交叉路口同时控制多辆车。5）我们还发布了用于训练和测试该方法的完整代码库。代码可在此处找到：https://github.com/Dekai21/Multi_Agent_Intersection。

请注意，本文的重点是图1第 3 步中的 MTP 模块，我们展示了如何利用多辆车辆的状态和意图信息，同时预测它们的未来轨迹。第 1 步涉及获取车辆的鸟瞰图信息，第 2 步涉及将意图信息传递至 MTP，这两步将在相关工作部分II中略作介绍。基于此，作出以下假设：

1) 可获取车辆的状态信息和意图。2) 在需要控制时，车辆能够接收控制指令，以便在交叉路口执行正确的机动操作。

II. 相关工作

无线车辆通信：
车联万物/车联基础设施（V2X/V2I）可与车辆进行无线通信 [21]。与安装在自车上的常规视距传感器相比 [22], V2X/V2I具有多种优势。它能够实现可靠的数据传输 [23] ，即使在附近存在非视距车辆时也能及时发出预警，提示即将发生的交通拥堵 [24], 、紧急制动 [25], 或危险超车[26]。在我们的框架中，需要向车辆传输状态信息，并接收来自车辆的控制指令。不过，我们的工作重点是多车辆轨迹预测，而非车辆通信。因此，我们假设可以通过实验中使用的不同轨迹数据集/平台获取意图信息。

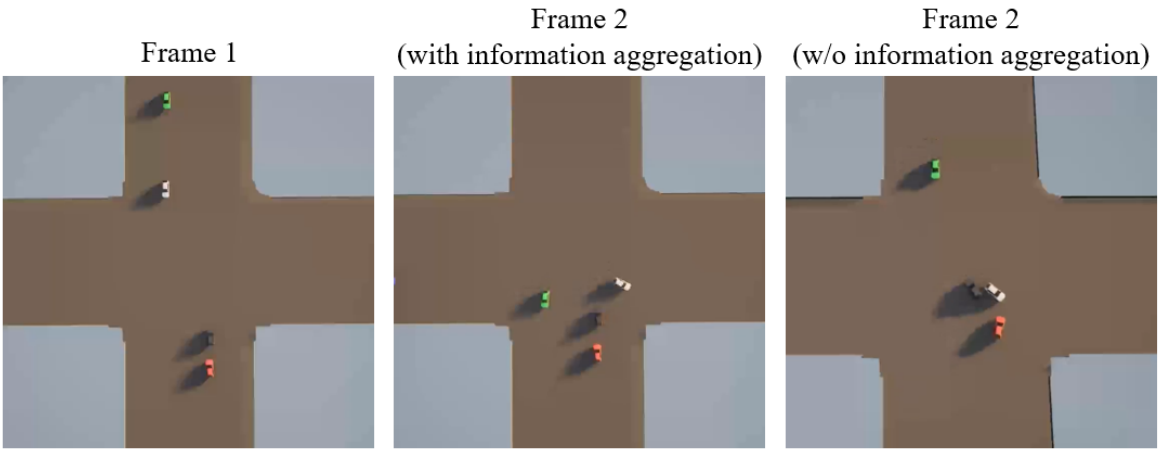


图 2：该图展示了信息聚合的重要性。左： 描述了一个由 4 辆车辆组成的初始场景。白色车辆是第一辆从上方驶抵交叉路口的车辆，并打算向左转。同时，从下方驶来的棕色和红色车辆分别打算左转和直行。中： 通过信息聚合，棕色和红色车辆会等待，直到白色车辆驶离交叉路口。右： 在无信息聚合的情况下，棕色和红色车辆会提前开始移动，并与白色车辆相撞。

轨迹数据集：
有多个真实世界数据集从鸟瞰视角提供各种道路参与者的轨迹信息。例如， [17], [18] 从建筑物顶部采集数据，而 [15], [16] 则利用无人机影像采集数据。不过，这些数据集中包含的车辆轨迹在场景中的占比有限，数量和精度都不足以训练数据驱动学习算法。因此，我们在真实世界的InD数据集 [19], 上进行训练和测试，以解决这些局限。请注意，我们的轨迹预测方法也可以扩展用于控制多辆车辆。这属于具身智能体评估[27] 的范畴，也是深度学习领域一个新兴的研究主题。不过，上述数据集都不支持在线评估 [28]。因此，我们使用城市交通仿真（SUMO）平台[29]。在本研究中，SUMO支持创建和控制交叉路口的各种场景，例如车辆数量/意图、道路优先级、交叉路口结构等。在SUMO上完成训练后，我们会在一个完全不同的平台——Car Learning to Act（CARLA） [30]上评估车辆的在线控制。CARLA支持向场景中的多辆车辆发送转向和加速/油门指令，以操控车辆。

多智能体轨迹预测：
未来轨迹预测之间社会交互信息的方法已应用于行人 [31], [32], [33], [34] 和车辆 [35], [36], [37], [38]。许多此类方法利用车辆的过去轨迹来推断未来 [39], [40]。在 [41], [42] 中，输出是概率性的；而在 [43], [44]中，输出则是多模态的，尤其是在道路分岔成多个方向的位置。然而，

如果车辆打算驶向某个特定方向，那么多个备选方案中只有一个是有效的。相比之下，我们的方法不需要过去轨迹信息，只需要车辆的当前状态。此外，由于我们的模型以车辆的意图为条件，未来轨迹的预测是唯一的。[45] 表明，了解其他车辆的意图有助于车辆在T形交叉路口汇入车流。掌握意图信息后，我们的轨迹预测任务还可扩展为控制车辆驶向目标位置。具体而言，我们通过应用模型预测控制来确定合适的油门和转向角，使车辆沿着预测轨迹行驶。据我们所知，尚无先前方法仅使用当前状态和意图信息来预测多辆车辆的未来轨迹并对其进行控制。 [46],近期的一项研究使用了状态和意图信息，但仅用于维持前后车辆之间的纵向安全距离。此外，该研究采用基于规则的方法，而我们的方法则通过训练神经网络实现数据驱动。

多智能体控制：
在交叉路口控制单辆车是一项复杂的任务 [47]。如果还需要处理与其他智能体的交互，任务就会更加棘手[48]。 [49], [50], [51], [52] 的工作旨在控制多辆车辆的通行，以减少交叉路口的交通拥堵和碰撞。不过，他们是通过控制交通信号灯来实现这一目标的。相比之下，我们的网络致力于控制没有交通信号灯的交叉路口中的单辆车辆。 [53], [54] 通过领航者引导的编队控制来处理多个智能体。而在我们的工作中，所有车辆都是独立控制的。其他控制单辆车的方法则需要求解优化问题 [55], [56]。相比之下，我们的方法以学习为基础

[57], [58], [59], [60] 采用强化学习（RL）进行多智能体预测/控制。然而，强化学习方法往往极其耗费数据 [61]。相比之下，我们的框架采用模仿学习，并辅以额外的碰撞代价，以防止同时控制多辆车辆时发生车辆间碰撞。

III. 框架

本节将介绍图中所示的多车辆轨迹（MTP）模块的具体细节。1。该模块以场景中每辆N 辆车的状态与意图信息作为输入，并预测它们未来T 个时间步的轨迹。我们的框架由以下部分组成：

输入：
输入到 MTP 的每辆车信息由向量 $X_k \in \mathbb{R}^6$、$k = 1,\ 2,$ .....N表示。X_k又由两个部分组成：1）状态；2）车辆的意图。状态：车辆k 的状态由向量 $\in \mathbb{R}^3$ 表示，该向量由车辆的朝向（$\theta_k \in \mathbb{R}$）以及其在x –y平面上的位置 $S_k \in \mathbb{R}^2$ 描述。意图：车辆k 的意图由$I_k \in \mathbb{R}^3$ 表示，是一个独热编码向量，用于描述车辆在即将到达的交叉路口是否打算左转、右转或直行。

输入转换：
随后，将每辆车的输入向量X_k 依次传入一系列具有可训练参数的L 个多层感知机（MLP）层。每辆车k 在l 层的输出是一个潜在表示，由$X_k^l \in \mathbb{R}^l$ 给出，具体方程如下：

$$X_k^l = \begin{cases} X_k & l = 0 \\ \sigma(\mathbf{W}^l X_k^{l-1} + \mathbf{b}^l) & 0 < l \le L \end{cases} \tag{1}$$

其中，$\mathbf{W}^l \in \mathbb{R}^{l\times(l-1)}$ 和$\mathbf{b}^l \in \mathbb{R}^l$ 是 MLP 第l层的可训练参数，而 σ 是 ReLU非线性激活函数。

信息聚合：
请注意，车辆k 的最后一个 MLP 层L 的输出是X_k^L，且仅取决于前一层中同一车辆的潜在表示。为了对车辆的未来轨迹做出充分知情的预测，除了考虑车辆自身的潜在信息外，也应考虑其他车辆的信息。因此，在后续层$l = L+1, L+2, ...L_F$中，通过消息传递进行信息聚合。由此得到每辆车的新潜在表示，如下式所示 [62]:

$$X_k^l = \begin{cases} X_k^L & l = L \\ \sigma(\mathbf{W}_s^l X_k^{l-1} + \mathbf{W}_o^l \sum_{p=1,p\neq k}^{N} X_p^{l-1}) & L < l < L_F \\ \mathbf{W}_s^l X_k^{l-1} + \mathbf{W}_o^l \sum_{p=1,p\neq k}^{N} X_p^{l-1} & l = L_F \end{cases} \tag{2}$$

其中，$\mathbf{W}_s^l,\ \mathbf{W}_o^l \in \mathbb{R}^{l\times(l-1)}$ 是聚合层的可训练参数。最后一层中每辆车k 的输出为$X^{L_F}{}_k \in \mathbb{R}^{2T}$。它表示对车辆 k 未来 T 个时间步的轨迹信息S_k 的预测。请注意，在实验中，我们展示了聚合相邻车辆信息的重要性。我们证明，当移除式（2 ）中表示聚合相邻节点信息的第二项时，训练后模型的性能会显著下降。

请注意，尽管各车辆共享权重，但每辆车预测的轨迹各不相同，因为每辆车的输入向量（由状态与意图信息构成）都不同。

损失函数：
用于训练模型的损失函数可分解为模仿损失（$L_{imitation}$）和碰撞损失（$L_{collision}$）。模仿损失是模型预测的未来轨迹与真实值之间L_2 距离的均值。

$$L_{imitation} = \frac{1}{N}\sum_{k=1}^{N}\sum_{t=1}^{T} |S_k^t - \hat{S}_k^t|_2 \tag{3}$$

ˆ其中，S_k^t 和S_k^t 分别是车辆k 在时间步t的预测状态信息和真实状态信息。同时，如果任意两辆车辆（例如车辆i和车辆j）的未来轨迹在同一时刻t相距小于某个安全距离阈值 λ，则会将与超出量成正比的碰撞代价计入碰撞损失：

$$L_{collision} = \sum_{i,j} L_{collision_{i,j}} \tag{4}$$

$$L_{collision_{i,j}} = \begin{cases} 0 & if \min_t |S_i^t - S_j^t|_2 > \lambda \\ \lambda - \min_t |S_i^t - S_j^t|_2 & otherwise \end{cases} \tag{5}$$

其中，$1 \le i < j \le N$ 和 $1 \le t \le T$。碰撞损失旨在降低交叉路口发生车辆间碰撞的倾向。我们在实验中展示了损失函数这一组成部分的重要性。

车辆控制：
请注意，我们的多车辆轨迹模块还可以扩展为控制各个车辆。为此，我们使用自行车模型 [63] 对汽车进行建模，并应用模型预测控制（MPC）来优化加速度(a)和转向角 (δ)，以跟随车辆预测轨迹上选定的J 个点

。与其他控制器 [64], [65]相比，MPC已证明具有更好的性能。采用自行车模型时的运动方程如下：

$$\dot{x} = v \cdot cos\theta;\ \dot{y} = v \cdot sin\theta;\ \dot{v} = a;\ \dot{\theta} = v\frac{\tan\delta}{L} \tag{6}$$

其中，L 为轴距，v 为车辆速度。同时，优化过程中最小化的代价函数如下：

$$\min_{a,\delta} \sum_{i=1}^{J} [(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 + (\theta_i - \hat{\theta}_i)^2] \tag{7}$$

数据增强：
请注意，在车辆在线控制中，仅使用记录的数据进行训练可能并不足够。这是因为控制车辆的参数可能导致自车偏离预期轨迹。这种偏离会使自车进入模型训练期间未见过的场景，例如对向交通所在的车道或道路边界。由于训练集中没有此类场景，模型对未来轨迹的预测会出错，进而导致控制参数使车辆进一步偏离正常轨迹，最终撞上侧边护栏。因此，为了防止发生此类护栏碰撞，我们还通过在车辆位置上添加噪声来增强原始记录数据。随后，使用式6 和 7所述的模型预测控制来确定输出的未来轨迹。不过，唯一的区别在于，优化仅针对轨迹上的最后一个点进行，而非针对已知轨迹上的 J 个点进行。实验表明，加入这种增强能够减少车辆控制过程中与护栏的碰撞。

IV. 实验

为衡量我们框架的性能，我们进行了离线和在线评估。离线评估用于评估训练好的模型对未来轨迹的预测能力。为此，我们使用真实世界的 inD数据集 [19]。

需要注意的是，我们的轨迹预测方法也可扩展用于控制车辆在交叉路口的行驶。不过，离线评估未必能反映真实的驾驶质量。事实上， [28]表明，离线指标相似的两个模型在实际部署时，性能可能存在巨大差异。因此，有必要进行在线评估，让智能体能够主动与环境交互。为此，我们使用 CARLA [30] 平台进行在线评估，而模型则是在另一个平台 SUMO [66]上训练的。SUMO-CARLA 联合仿真为这项评估提供了支持。

A. 离线评估：

图中所示的来自 inD 数据集的 Bendplatz 和弗兰肯堡交叉路口3 已用于离线评估。对于每个交叉路口，随机选取 3 个轨迹文件用于训练

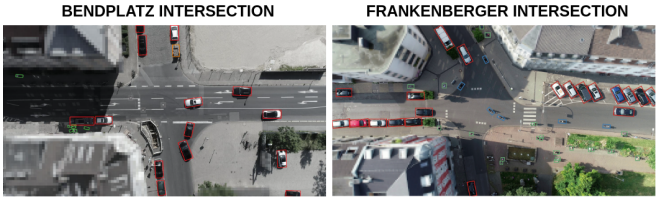


图 3：德国 Bendplatz 和 Frankenburg 交叉路口在 inD 数据集中的图像快照 [19]

和 1 个用于验证。每个轨迹文件包含在不同时间采集的 20 分钟轨迹记录。采用 4 个常用的离线评估指标进行比较，分别为：平均位移误差（ADE）、最终位移误差（FDE）、漏检率（MR）和碰撞率（CR） [39], [67]。对于感兴趣的读者，这些指标的数学公式与解释，以及有关实验中使用的 inD 数据集的更多信息，详见补充文件[1]。

除我们的模型外，我们还训练了另外 3 个模型以供比较。下面介绍这些模型：

过去轨迹（VectorNet）：
该模型改编自 [39] 的方法。与我们的方法类似，它会获取周围车辆的信息，并使用基于注意力的机制来实现这一点。不过，该方法还会使用车辆当前状态以及过去轨迹的信息，以便更准确地判断未来轨迹。相比之下，我们的模型使用车辆想要前往何处的意图，而不是过去轨迹。

无信息聚合：
该模型的架构与我们的模型相似，不同之处在于车辆不会聚合环境中其他车辆的信息。为此，在进行轨迹预测时会阻止车辆之间的消息传递。此外，训练时仅使用模仿学习损失。

无碰撞代价：
这也与我们的方法类似，不同之处在于，该模型的训练未采用我们在模仿学习范式中引入的额外碰撞代价。

我们的方法：
该模型采用第 III 节所述的框架进行训练。模型获取场景中每辆车辆的状态和意图信息，并基于这些可用信息预测每辆车辆的未来轨迹。请注意，该模型能够通过消息传递，让每辆车辆汇总其他车辆的信息。这种对环境的整体表示应有助于做出信息充分的轨迹预测，从而最大限度地减少多个智能体之间的碰撞。模型使用以下两种损失函数进行训练：

[1]https://github.com/Dekai21/Multi_Agent_Intersection/tree/master/supplementary

模仿损失函数和碰撞损失函数。不过，请注意，此处未采用旨在车辆在线控制的数据增强。

所有 4 个模型的离线评估结果见表I和表II。

B. 在线评估/控制

在 CARLA 平台上对驾驶质量进行在线评估。然而，模型是在 SUMO 数据上训练的。交叉路口的设计使得纵向道路（南北向）的优先级高于横向道路（东西向）。用于评估在线驾驶质量的指标是距离碰撞比（*DCR*）。这是一项在线指标，用于描述智能体在发生车辆间（V2V）或车辆与障碍物（V2B）碰撞之前行驶的距离。其数学定义为交叉路口所有车辆行驶的总距离除以发生的 V2V 或 V2B 碰撞总次数。该指标的值越高越好。

$$DCR = \frac{1}{C}\sum_{k=1}^{N}\sum_{t=1}^{T_k-1}\sqrt{(x_{t+1,k}-x_{t,k})^2+(y_{t+1,k}-y_{t,k})^2} \quad (8)$$

其中，C 表示 V2V 或 V2B 碰撞的次数。同时，T_k 表示车辆k 通过交叉路口所需的时间步数。通常，左转车辆所需的时间步数多于右转车辆，因为相应曲线的周长不同。

用于本次在线评估的模型与离线评估小节 IV-A 中所述的模型相同。唯一的区别是，我们额外训练了两个采用数据增强的模型，以增强其对不精确预测所引起偏差的鲁棒性。第一个模型采用数据增强训练，但不使用碰撞损失；另一个模型则同时采用数据增强和碰撞损失进行训练。所有这些模型的 V2V 和 V2B 碰撞的 DCR指标见表 III。为便于复现，在线控制的推理代码以及 SUMO-CARLA 联合仿真的设置详情均已提供于以下代码仓库：https://github.com/Dekai21/Multi_Agent_Intersection#run-the-inference-code。

C. 讨论：

本小节将进一步阐述从结果中得出的一些发现。

聚合的重要性：
可以看出，不聚合其他车辆信息的模型表现不如我们的模型。这是因为交叉路口是多个车辆之间会发生大量交互的位置。因此，如果不进行聚合，智能体只能接收到关于自身的信息，而无法获知其他车辆的状态、意图和行为。因此，它无法全面审视整个

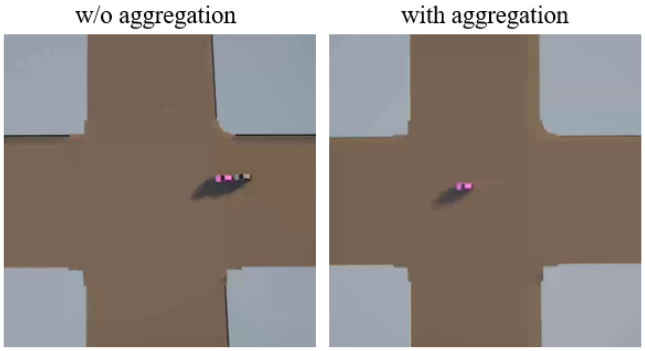


图 4：展示了在 CARLA 仿真器的一个交叉路口中，与采用聚合的模型相比，不使用聚合会产生何种影响的示例。水平车道（左右方向）为非优先道路，垂直车道（上下方向）为优先道路。

场景，也就无法在做出明智决策时准确预测自身轨迹。在 CARLA 上进行在线评估时，我们观察到一个有趣的现象：大多数碰撞并非发生在交叉路口内，而是在车辆从非优先道路驶入交叉路口之前。这是因为在训练集中，这些车辆会通过减速甚至完全停车，在进入交叉路口前让行给优先道路上的车辆，以便优先道路上的车辆顺利通行。只有在不会妨碍其他车辆时，非优先道路上的车辆才会驶入交叉路口。然而，与非优先道路上的车辆静止不动的样本数量相比，这类情况十分罕见。因此，如果无法获取其他智能体的信息，模型就会记住从非优先道路驶入交叉路口前始终保持静止。这会阻塞非优先道路，妨碍其他车辆通行。在理想情况下，如果有车辆阻塞道路，驶近这一瓶颈点的其他车辆理应减速以避免碰撞。然而，这些车辆同样无法察觉阻塞道路的车辆，仍试图从中穿过，从而导致碰撞。这尤其会拉低 DCR指标，特别是车对车碰撞方面。

图4 展示了未进行信息聚合所造成的后果。如前所述，由于第一辆静止的车辆，其他车辆在进入交叉路口之前发生了碰撞。同一图的右侧展示了我们采用聚合的模型所生成的一个示例场景。粉色车辆希望直行，它知道交叉路口内没有其他车辆，因此驶入了交叉路口。

碰撞代价的贡献：
可以观察到，在训练过程中使用碰撞成本惩罚的模型，性能优于未使用该惩罚进行训练的模型。在线指标上的效果尤为显著，特别是在防止车对车碰撞方面。请注意，移除这一损失项后，V2V 的 DCR 指标会

表 I：轨迹预测在 InD数据集的 Bendplatz交叉路口处的结果。（指标值越低越好r)

模型	Past 轨迹	意图	消息 超车	碰撞 Cost	ADE	FDE	MR	CR	漏检率+碰撞率
Past 轨迹	✓		✓		3.800	7.515	0.816	0.043	0.859
无信息聚合		✓			1.341	2.619	0.230	0.127	0.357
No 碰撞代价		✓	✓		1.110	2.193	0.172	0.101	0.273
Our 模型		✓	✓	✓	1.099	2.126	0.157	0.075	**0.232**

表 II：InD数据集中弗兰肯堡交叉路口的轨迹预测结果。（指标值越低越好）

模型	Past 轨迹	意图	消息 超车	碰撞 Cost	ADE	FDE	MR	CR	漏检率+碰撞率
Past 轨迹	✓		✓		2.192	4.437	0.513	0.092	0.605
无信息聚合		✓			1.958	3.924	0.411	0.147	0.558
No 碰撞代价		✓	✓		1.752	3.518	0.341	0.112	0.453
Our 模型		✓	✓	✓	1.850	3.623	0.359	0.072	**0.431**

表 III：CARLA 在线评估结果。（指标值越高越好）

模型	Past 轨迹	意图	消息 超车	碰撞 Cost	MPC Aug.	DCR（车车）	DCR（车路）
Past 轨迹	✓		✓			99.1	158.6
无信息聚合		✓				168.7	607.4
无碰撞代价 & 增强		✓	✓			722.2	515.9
No 增强		✓	✓	✓		925.2	341.3
No 碰撞代价		✓	✓		✓	753.6	1256.0
Our 模型		✓	✓	✓	✓	**3915.0**	**1957.5**

大幅下降。碰撞代价的作用在于，当模型感知到潜在碰撞时，能够对车辆轨迹进行细微调整，从而避开其他车辆。补充材料中包含一段视频，展示了不使用碰撞代价与采用我们的方法所产生的差异。

数据增强的重要性：
请注意，我们引入了数据增强，以防止车辆在在线评估期间偏离道路并撞上路障。与未使用数据增强训练的模型相比，碰撞指标显著降低，尤其是车辆与障碍物碰撞率。相比之下，我们的模型使用了处于偏离正常轨迹位置的数据样本进行训练，因此，即使模型在在线推理期间最终到达偏离轨迹的位置，也能知道应采取何种纠正措施，使车辆回到正轨。如果路障或其他车辆挡在前方，这样就能避免与它们发生碰撞。

过去轨迹信息：
请回想一下， [39] 中的模型使用车辆的过去轨迹信息来预测未来轨迹。因此，这类模型具有概率解释：精确的未来轨迹往往模糊不清，直到车辆深入交叉路口后才逐渐变得更加明确。相比之下，由于我们的模型获得了车辆意图的信息，其预测结果是唯一且准确得多的，结果也印证了这一点。这种意图使我们的方法能够

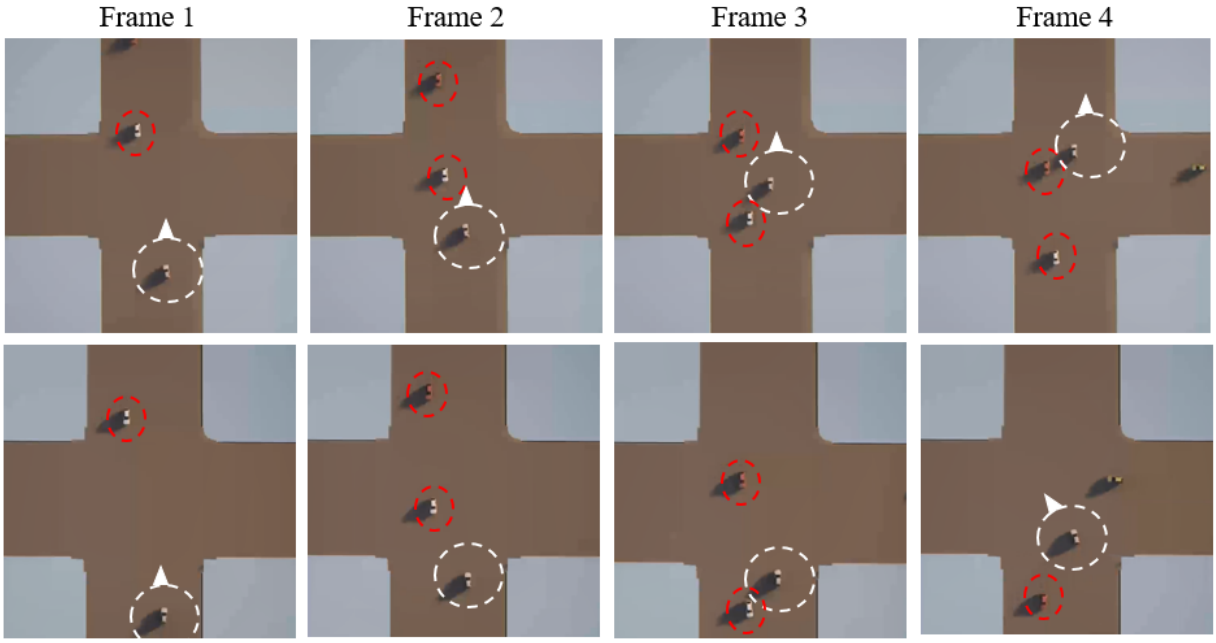


图 5：展示了如何利用意图控制车辆之间的行为与交互。在第一排图像中，来自下方、被白色圆圈标出的车辆想要直行。它持续行驶，不让行于其他车辆。在第二排中，同一辆车的意图被修改为左转。此时，白色车辆减速，让行于被红色圆圈标出的直行车辆。红色车辆通过后，白色车辆才开始执行左转。白色圆圈上的箭头表示行驶方向。若被白色圆圈标出的车辆处于静止状态，则不会显示箭头。请注意，来自右侧的车辆打算右转，因此当白色圆圈标出的车辆打算左转时，它不会构成阻碍。

扩展到车辆控制。图 5 显示，通过操纵意图，可以相应地调整车辆之间的交互。这种改变行为的灵活性，只有在输入中使用车辆意图所带来的能力支持下才能实现。与使用过去轨迹信息相比，离线轨迹预测不仅更加准确（见表I 和II ），在线控制也更加稳健（见表 III）。

领域自适应：
请注意，我们的模型仅使用来自 SUMO 平台的数据进行训练，以预测未来轨迹，但它也可以用于控制完全不同平台上的车辆。本例中，该平台为 CARLA。请注意，在训练期间，模型无法获取来自 CARLA 的数据。模型能够成功适应不同领域，是因为我们使用车辆的状态信息作为表征。这种表征在不同平台／领域之间保持一致。因此，模型不受该表征来源的影响，即无论来源是 CARLA 还是 SUMO。图像等其他表征则难以在不同领域、天气／光照条件等之间切换。例如，在晴天条件下使用图像训练的控制模型，即使领域相同，在雨天条件下也难以控制车辆 [68]。

请注意，用于训练以及进行离线和在线评估的完整代码包含在以下文档中：以下代码仓库：https://github.com/Dekai21/Multi_智能体_交叉路口。

V. 结论

本文展示了如何利用多辆车辆的状态与意图信息，同时预测它们在交叉路口的轨迹。借助这一方法，还可以进一步控制车辆驶向预期方向。汇总其他车辆的信息，也有助于每辆车辆做出更明智的决策。我们的框架还能够在一个领域的数据上进行训练，并在另一个领域的数据上测试，而后者的数据在训练期间并未见过。

参考文献

[1] SAE国际，“道路机动车辆驾驶自动化系统相关术语的分类与定义”，*SAE*国际，2021年。[2] Kadir Korkmaz，“利用自动驾驶车辆GPS数据和卡尔曼滤波器生成位置信息”，载于*2017*年第*25*届信号处理与通信应用会议（*SIU*），2017年。[3] Shaojiang Zhang、Yanning Guo、Qiang Zhu和Zhiyuan Liu，“基于激光雷达、惯性测量单元和车轮里程计的自动驾驶车辆定位系统”，载于中国控制与决策会议（*CCDC*），2019。[4] Hermosa Almeida等，“面向户外应用，采用低成本惯性测量单元/全球定位系统集成的小型地面车辆自主导航”，载于*IEEE*国际系统会议（*SysCon*），2019。[5] Ankith Manjunath等，“面向自动驾驶的基于雷达的目标检测与跟踪”，载于*IEEE*微波理论与技术学会智能移动国际会议（*ICMIM*），2018。

[6] Myeon-gyun Cho，“使用激光雷达和热红外摄像机进行自动驾驶遥控汽车障碍物识别的研究”，载于第十一届普适与未来网络国际会议（ICUFN），2019。[7] Heong-tae Kim和Bongsob Song，“基于雷达与视觉传感器融合的车辆识别，用于自动紧急制动”，载于2013年第十三届控制、自动化与系统国际会议（ICCAS 2013），2013，第1342–1346页。[8] Jinghua Guo、Ping Hu和Rongben Wang，“紧急避障中基于视觉的自动驾驶车辆非线性转向与制动协调控制”，《IEEE智能交通系统汇刊》，第17卷，第11期，第3230–3240页，2016年。[9] Farzeen Munir等，“Ldnet：一种使用动态视觉传感器的端到端车道线检测方法”，《IEEE智能交通系统汇刊》，第23卷，第7期，2022年。[10] Zeng Li、Shaosong Li、Zheng Li、Gaojian Cui和Xiaodong Wu，“基于视觉导航的智能车辆侧风条件下车道保持”，载于2018年第五届信息科学与控制工程国际会议（ICISCE），2018，第290–294页。[11] Aleksandr Kim、Aljosa Osep和Laura Leal-Taixe，“Eagermot：通过传感器融合实现三维多目标跟踪”，载于IEEE机器人与自动化国际会议（ICRA）。IEEE，2021年。[12] Tim Meinhardt、Alexander Kirillov、Laura Leal-Taixe和Christoph Feichtenhofer，“Trackformer：基于Transformer的多目标跟踪”，载于IEEE计算机视觉与模式识别会议（CVPR），2022年6月。[13] R. Marzoug、N. Lakouari、H. Ez-Zahraouy、B. Castillo T´ellez、M. Castillo T´ellez和L. Cisneros Villalobos，“利用元胞自动机对信号交叉口的汽车事故进行建模与仿真”，《物理学A：统计力学及其应用》，第589卷，第126599页，2022年。[14] Gang-Len Chang和Hua Xiang，“拥堵程度与事故之间的关系”，技术报告，州公路管理局，2003年。[15] Alexandre Robicquet、Amir Sadeghian、Alexandre Alahi和Silvio Savarese，“学习社会交往礼仪：拥挤场景中的人体轨迹理解”，载于《计算机视觉——ECCV 2016》，Bastian Leibe、Jiri Matas、Nicu Sebe和Max Welling编，Cham，2016，第549–565页，Springer国际出版公司。[16] Dongfang Yang等，“俯视轨迹：通过受控实验和拥挤的校园采集的车辆与人群交互行人数据集”，载于2019年IEEE智能车辆研讨会（IV），2019年。[17] Alon Lerner、Yiorgos Chrysanthou和Dani Lischinski，“Crowds by example”，《计算机图形论坛》，第26卷，2007年。[18] S. Pellegrini、A. Ess、K. Schindler和L. van Gool，“你永远不会独行：面向多目标跟踪的社会行为建模”，载于2009年IEEE第十二届计算机视觉国际会议，2009年。[19] Julian Bock、Robert Krajewski、Tobias Moers、Steffen Runde、Lennart Vater和Lutz Eckstein，“InD数据集：德国交叉路口自然交通参与者轨迹的无人机数据集”，载于2020年IEEE智能车辆研讨会（IV），2020年，第1929–1934页。[20] Justin Gilmer等，“用于量子化学的神经消息传递”，载于机器学习国际会议。PMLR，2017年。[21] Meriem Houmer、Mariyam Ouaissa和Mariya Ouaissa，“基于5G的V2X通信安全认证方案”，《计算机科学程序汇刊》，第198卷，第276–281页，2022年。[22] Fabian de Ponte M¨uller、Estefania Munoz Diaz和Ibrahim Rashdan，“用于车辆相对定位的协同定位与雷达传感器融合”，载于2016年IEEE智能车辆研讨会（IV），2016年，第1060–1065页。[23] Jean-Philippe Vasseur和Adam Dunkels，“第22章——智慧城市与城市网络”，载于《通过IP互联智能物体》，Jean-Philippe Vasseur和Adam Dunkels编，第335–351页。Morgan Kaufmann，波士顿，2010年。[24] Ignacio Llatser、Thomas Michalke、Maxim Dolgov、Florian Wild-sch¨utte和Hendrik Fuchs，“面向5G V2X通信的协同自动驾驶应用场景”，载于2019年IEEE第二届5G世界论坛（5GWF），2019年，第120–125页。[25] Feng Zhao和Leonidas J. Guibas，“8——应用与未来方向”，载于《无线传感器网络》，Feng Zhao和Leonidas J. Guibas编，Morgan Kaufmann网络系列，第291–306页。Morgan Kaufmann，旧金山，2004年。[26] Ruoqi Deng、Boya Di和Lingyang Song，“基于蜂窝V2X的自动驾驶车辆超车动作中的协同避碰驾驶，”IEEE车辆技术汇刊，第68卷，第5期，第4434–4446页，2019年。[27] Peter Anderson、Angel X. Chang、Devendra Singh Chaplot、Alexey Dosovitskiy、Saurabh Gupta、Vladlen Koltun、Jana Kosecka、Jiten- dra Malik、Roozbeh Mottaghi、Manolis Savva和Amir R. Za- mir，“具身导航智能体的评估”，CoRR，第abs/1807.06757卷，2018年。[28] Felipe Codevilla、Antonio M Lopez、Vladlen Koltun和Alexey Dosovitskiy，“基于视觉的驾驶模型的离线评估”，载于《欧洲计算机视觉会议（ECCV）论文集》，2018年，第236–251页。[29] Pablo Alvarez Lopez等，“使用sumo进行微观交通仿真”，载于《2018年第21届国际智能交通系统会议（ITSC）》，2018年，第2575–2582页。[30] Alexey Dosovitskiy、German Ros、Felipe Codevilla、Antonio Lopez和Vladlen Koltun，“CARLA：一个开放的城市驾驶模拟器”，载于《第一届机器人学习年会论文集》，2017年。[31] Haowen Tang、Ping Wei、Jiapeng Li和Nanning Zheng，“Evost-gat：面向行人轨迹预测的时空图注意力网络演化”，Neurocomputing，第491卷，第333–342页，2022年。[32] Yusheng Peng、Gaofeng Zhang、Jun Shi、Benzhu Xu和Liping Zheng，“Srai-lstm：一种基于社会关系注意力的交互感知LSTM，用于人类轨迹预测”，Neurocomputing，第490卷，第258–268页，2022年。[33] Hao Zhou、Dongchun Ren、Huaxia Xia、Mingyu Fan、Xu Yang和Hai Huang，“Ast-gnn：一种用于交互感知行人轨迹预测的基于注意力的时空图神经网络”，Neurocomputing，第445卷，第298–308页，2021年。[34] Fang Zheng等，“异构轨迹预测中的无限邻域交互”，载于《IEEE/CVF国际计算机视觉会议论文集》，2021年。[35] Xiaoyu Mo、Zhiyu Huang、Yang Xing和Chen Lv，“采用异构边增强图注意力网络的多智能体轨迹预测”，IEEE智能交通系统汇刊，第23卷，第7期，第9554–9567页，2022年。[36] Hengbo Ma、Yaofeng Sun、Jiachen Li和Masayoshi Tomizuka，“利用自监督领域知识实现跨不同场景的多智能体驾驶行为预测”，载于《2021年IEEE国际智能交通系统会议（ITSC）》，2021年。[37] Tianyang Zhao、Yifei Xu、Mathew Monfort、Wongun Choi、Chris Baker、Yibiao Zhao、Yizhou Wang和Ying Nian Wu，“用于上下文轨迹预测的多智能体张量融合”，载于《IEEE/CVF计算机视觉与模式识别会议（CVPR）论文集》，2019年6月。[38] Xin Li等，“Grip：基于图的交互感知轨迹预测”，载于《IEEE智能交通系统会议》，2019年。[39] Jiyang Gao等，“Vectornet：从向量化表示中编码高精地图和智能体动态”，载于《IEEE/CVF计算机视觉与模式识别会议论文集》，2020年。[40] Xiaosong Jia等，“结合自我中心视角与客体中心视角的多智能体轨迹预测”，载于《第五届机器人学习会议论文集》，2022年，PMLR。[41] Jiachen Li等，“通过对抗学习实现交互感知多智能体跟踪与概率行为预测”，载于《国际机器人与自动化会议（ICRA）》，IEEE，2019年。[42] Hengbo Ma、Jiachen Li、Wei Zhan和Masayoshi Tomizuka，“具有运动学约束的Wasserstein生成学习，用于概率交互式驾驶行为预测”，载于《2019年IEEE智能车辆研讨会（IV）》，IEEE，2019年，第2477–2483页。[43] NN Sriram、Buyu Liu、Francesco Pittaluga和Manmohan Chan- draker，“Smart：同时进行多智能体循环轨迹预测”，载于《欧洲计算机视觉会议》，Springer，2020年。[44] Francesco Marchetti、Federico Becattini、Lorenzo Seidenari和Al- berto Del Bimbo，“利用记忆增强网络预测运动智能体的多条轨迹”，IEEE模式分析与机器智能汇刊，第1–1页，2020年。[45] Volkan Sezer、Tirthankar Bandyopadhyay、Daniela Rus、Emilio Fraz- zoli和David Hsu，“面向人类驾驶员意图不确定情况下无信号交叉路口的自主导航”，载于《2015年IEEE/RSJ国际智能机器人与系统会议（IROS）》，2015年，第3578–3585页。[46] Hao M. Wang、Sergei S. Avedisov、Onur Altintas和G´abor Orosz，“考虑时间延迟的状态与意图共享式多车辆冲突管理”，IEEE智能车辆汇刊，第1–14页，2022年。

[47] Zi-jia Wang、Xue-mei Chen、Pin Wang、Meng-xi Li、Han Zhang等，“一种基于冲突解决的城市交叉路口自动驾驶车辆决策模型”，《先进交通杂志》，第2021卷，2021年。[48] Qiang Ge、Qi Sun、Zhen Wang、Shengbo Eben Li、Ziqing Gu、Sifa Zheng和Lyuchao Liao，“利用图形混合整数优化实现交叉路口网联车辆实时协调”，《IET智能交通系统》，第15卷，第6期，第795–807页，2021年。[49] Bo Liu和Zhengtao Ding，“一种用于交通信号灯控制的分布式深度强化学习方法”，Neurocomputing，第490卷，第390–399页，2022年。[50] Zhengyi Ge，“基于强化学习的信号控制策略，以提高城市交叉路口的通行效率”，载于《2020年城市工程与管理科学国际会议（ICUEMS）》，2020年，第347–351页。[51] Maheen Firdous、Fasih Ud Din Iqbal、Nouman Ghafoor、Nauman Khalid Qureshi和Noman Naseer，“使用模糊逻辑控制四岔路口和丁字路口交通信号灯的系统”，载于《2019年IEEE人工智能与计算机应用国际会议（ICAICA）》，2019年，第178–182页。[52] Mengyu Guo、Pin Wang、Ching-Yao Chan和Sid Askary，“一种用于城市交叉路口智能交通信号控制的强化学习方法”，载于《2019年IEEE智能交通系统会议（ITSC）》，2019年，第4242–4247页。[53] Sathishkumar Moorthy和Young Hoon Joo，“通过仿生神经动力学方法实现多个非完整移动机器人的分布式领航-跟随编队控制”，Neurocomputing，第492卷，第308–321页，2022年。[54] Shude He、Rourou Xu、Zhijia Zhao和Tao Zou，“具有可见性和性能约束的多辆自动驾驶车辆视觉神经编队跟踪控制”，Neurocomputing，第492卷，第651–663页，2022年。[55] Michael W Levin和David Rey，“自动驾驶车辆交叉路口控制的冲突点公式化方法”，《交通研究C辑：新兴技术》，第85卷，第528–547页，2017年。[56] Maximilian Kloock等，“多车辆的分布式模型预测交叉路口控制”，载于《IEEE智能交通系统会议（ITSC）》，IEEE，2019年。[57] Di Wang、Hongbin Deng和Zhenhua Pan，“Mrcdrl：基于深度强化学习的多机器人协调”，Neurocomputing，第406卷，第68–76页，2020年。[58] Praveen Palanisamy，“基于深度强化学习的多智能体网联自动驾驶”，载于《2020年国际联合神经网络会议（IJCNN）》，IEEE，2020年，第1–7页。[59] Xianjie Zhang、Yu Liu、Xiujuan Xu、Qiong Huang、Hangyu Mao和Anil Carie，“用于多智能体强化学习的结构关系推理演员-评论家”，Neurocomputing，第459卷，第383–394页，2021年。[60] David Sim˜oes、Nuno Lau和Lu´ıs Paulo Reis，“具有通信功能的多智能体集中式演员-评论家”，Neurocomputing，第390卷，第40–56页，2020年。[61] Nelson Vithayathil Varghese和Qusay H. Mahmoud，“多任务深度强化学习综述”，《Electronics》，第9卷，2020年。[62] Christopher Morris等，“Weisfeiler和Leman走向神经网络：更高阶图神经网络”，载于《AAAI人工智能会议》，2019年。[63] Danwei Wang和Feng Q，“四轮转向车辆的轨迹规划”，载于《IEEE机器人与自动化国际会议（ICRA）》，2001年。[64] Jia Liu等，“自动驾驶车辆中纯追踪、Stanley、LQR和MPC控制器的仿真性能评估”，载于《IEEE实时计算与机器人国际会议》，2021年。[65] Moveh Samuel等，“使用比例积分微分（PID）和模型预测控制（MPC）进行车道保持机动操作”，《机器人与控制杂志（JRC）》，第2卷，第2期，第78–82页，2021年。[66] Pablo Alvarez Lopez、Michael Behrisch、Laura Bieker-Walz、Jakob Erdmann、Yun-Pang Fl¨otter¨od、Robert Hilbrich、Leonhard L¨ucken、Johannes Rummel、Peter Wagner和Evamarie Wießner，“使用sumo进行微观交通仿真”，载于《第21届IEEE智能交通系统国际会议》，2018年，IEEE。[67] Wei Zhan等，“INTERACTION数据集：一个用于交互式驾驶场景、配有语义地图的国际性、对抗性和协作性运动数据集”，arXiv:1910.*03088 [*cs，eess*]*, 2019。[68] Q. Khan、P. Wenzel、D. Cremers和L. Leal-Taix´e，“面向跨天气条件泛化的感知运动控制”，载于*IEEE/RSJ*国际智能机器人与系统会议（*IROS*），2019。