arXiv:2304.00843v1 [cs.RO] 3 Apr 2023

一种基于交互式时空走廊的分层多车辆协同运动规划方法

Xiang Zhang、Boyang Wang、Yaomin Lu、Haiou Liu、Jianwei Gong 和 Huiyan Chen

摘要——在非完整动力学约束下，如何安全、高效地解决多车辆协同运动规划中的冲突一直是一项挑战。为多辆车构建时空走廊，可以解耦高维冲突，并进一步降低获取可行轨迹的难度。因此，本文提出一种基于交互式时空走廊（ISTC）的新型分层方法。在第一层中，基于初始引导轨迹，采用混合整数二次规划构建 ISTC，以解决一般多车辆场景中的冲突；随后，在第二层中，通过非线性规划生成满足车辆动力学约束的走廊内轨迹。引入 ISTC 后，多车辆协同运动规划问题能够解耦为单车轨迹优化问题，从而大幅分散计算压力，并在实际应用中具有巨大潜力。此外，所提方法在三维$(x,\ y,\ t)$构型空间中搜索可行解，相较于传统的速度-路径解耦方法，保留了更多可能性。通过在无信号交叉口和复杂密集场景中开展仿真实验，验证了所提框架的可行性和适应性。

关键词—**多车辆协调、运动规划、智能车辆、优化方法。**

I. 引言

多ULTI-VEHICLE 协作技术在交通、港口、仓储及其他场景中具有潜在价值，因此正受到越来越多的关注。车联网（V2X）技术的快速发展使车辆能够与周边车辆、基础设施和指挥终端进行信息交互[1]。在此基础上，当各车辆的状态和意图实现共享后，多车辆规划问题便不再需要独立求解。指挥终端可以对多辆车的行为进行全局协调，然后向相应的单辆车发送指令。

多车辆协同运动规划面临的最大挑战，是如何确保所有车辆在

本工作部分得到中国国家自然科学基金项目（52172378）和北京理工大学青年学者基金项目的资助。（通讯作者：Boyang Wang）
Xiang Zhang、Boyang Wang、Yaomin Lu、Haiou Liu、Jianwei Gong 和 Huiyan Chen 均就职于中国北京市 100081 北京理工大学机械工程学院（电子邮箱：xiangzhang0825@163.com、boyangwang@bit.edu.cn、luyaomin9612@163.com、bit lho@bit.edu.cn、gongjianwei@bit.edu.cn、Chen h y@263.net）

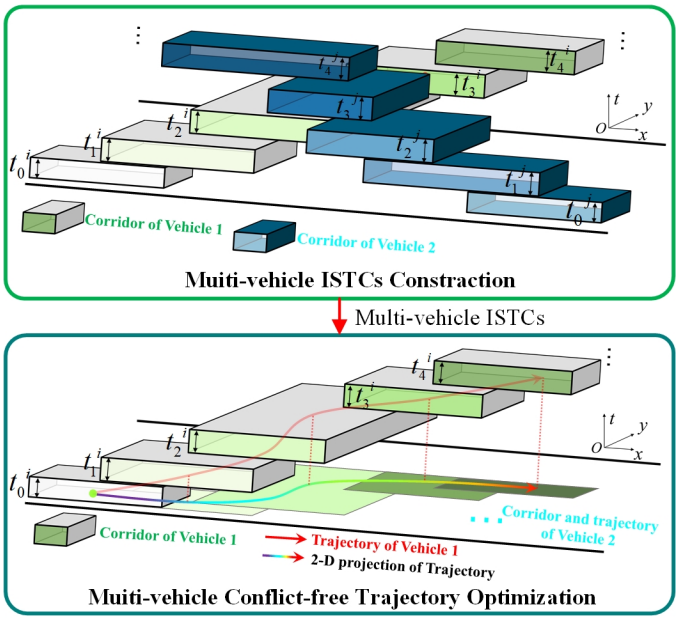


图1. 所提出的分层方法流程图。

时空构型空间中互不冲突。目前关于多车辆规划的研究主要聚焦于特定场景 [2], , 例如汇入和交叉路口。在常见场景中确保多车辆行驶安全高效的通用方法仍然缺乏。为改变这一现状，本文提出了一种基于交互式时空走廊（ISTC）的分层规划方法。交互式时空走廊为车辆及其他环境要素提供统一的表示，并在确保安全的同时高效地避免冲突。

如图1所示，本文提出了一种新颖的分层框架，以实现一般场景下的多车辆协同运动规划。在主框架的第一层中，设计了混合整数二次规划（MIQP），用于构建能够解决一般多车辆场景中冲突的 ISTC。ISTC 由所有车辆互不重叠的时空走廊组成，每辆车的走廊由一系列连续时间单位上的走廊立方体构成。随后，在第二层中，采用非线性规划（NLP）生成满足车辆动力学的走廊内轨迹。

本文的主要贡献如下：

- A 用于处理问题的新颖分层框架

提出了车辆协同运动规划方法。该框架可将多车辆协同运动规划问题解耦为满足指定约束的单车轨迹优化问题，从而降低整体复杂度，并提高一般场景下的效率

- 建立交互式时空耦合走廊，以实现时空域内各车辆的无冲突分区。通过在混合整数二次规划中设置适当的约束和优化目标，可有效解决多车辆场景中的强非凸冲突消解问题。

- 基于生成的交互式时空耦合走廊，各车辆在解耦的时空域内独立且最优地生成期望轨迹。最优轨迹生成会考虑走廊边界约束和车辆运动特性约束，并在冲突解耦空间中生成高质量的轨迹时间点。

本文其余部分安排如下：第二节介绍文献综述及相关工作。第三节描述所提算法的框架。第四节介绍面向非结构化环境的初始引导轨迹生成、ISTC的构建以及最终轨迹生成。第五节展示并讨论ISTC的结果以及最终轨迹仿真。最后，第六节给出结论和未来工作。

II. 相关工作

本节回顾机器人和车辆领域中协同运动规划的当前研究现状，并指出现有方法的不足。此外，还概述时空走廊在轨迹优化中的应用。

A. 移动机器人多智能体路径规划

如何为多机器人系统寻找无碰撞轨迹，被视为多智能体路径规划（MAPF）问题，多年来一直受到广泛研究。作为一个 NP 难问题 [3], ，人们提出了各种方法来为每个参与者规划轨迹，使其能够有效到达目标。解耦方法通常计算时间最短，其典型做法是每次选择一个智能体规划路径，确保所选智能体避开与其他智能体规划方案之间的冲突 [4]–[6]。然而，解耦导致解空间大幅缩减，在真实场景中难以接受。为缓解这一问题，人们采用基于规则的方法 [7],、A* [8]–[10], 以及冲突基搜索（CBS） [11]–[13], 同时为多个智能体寻找路径，以获得渐近最优解。近年来，基于学习的方法也广泛应用于多智能体路径规划。例如，Reijnen 等人使用强化学习（RL）获取启发式值 [14], ；Sinkar 等人采用分布式深度学习（DL）模型处理动态物体 [15] ； Bai 等人则结合长短期记忆（LSTM）与深度强化学习（DRL）以实现安全的编队控制 [16]。然而，尽管上述方法在简单任务场景中能够高效生成多智能体轨迹，但其中大多数方法大幅简化甚至完全忽略了车辆模型，也很少考虑交通场景的约束。因此，若要将这些方法应用于道路上的真实车辆，就应考虑更多车辆特性和环境约束方面的细节。

B. 多车辆协同运动规划

多车辆协同运动规划已在各种场景中得到广泛研究。在交叉路口场景中，基于多车辆编队控制 [17],半随机势场 [18], 和部分可观测马尔可夫决策过程（POMDP） [19] 的方法被用于实现车辆协作。在换道场景中，强化学习（RL）结合对手建模网络被用于实现高效执行 [20]。Li等人将换道机动分为两个阶段进行分布式规划，以平衡计算速度和质量 [21]。在超车场景中，基于人工势场法并结合编队控制的方法 [22] 以及基于距离估计策略的协同避碰方案 [23]被用于保障车辆安全。此外，为解决匝道协同问题，有研究提出结合人工神经网络（ANN）与深度强化学习（DRL）来计算纵向加速度 [24], ；也有研究整合控制障碍函数（CBFs）和控制李雅普诺夫函数（CLFs）以实现分布式控制 [25]，并对SAT（可满足性）求解器 [26] 进行了测试，取得了良好结果。然而，上述大多数方法都是针对特定交通场景设计的，无法应用于通用的多车辆交互环境。为在一般场景中为多辆车寻找无冲突轨迹，基于交互基元树的方法通过求解混合整数线性规划（MILP）来寻找最优协同模式 [27], [28]。但该方法的一个新缺陷是，对运动空间进行离散化会缩小解空间，并降低规划结果的最优性。

C. 基于时空走廊的轨迹优化

时空走廊本质上是车辆在整个时间轴上各个时刻可通行区域的集合。与在二维（x， y）平面中搜索的路径与速度解耦规划方法相比，采用时空走廊的规划方法可以在更大的三维（x， y， t）构型空间中搜索解决方案，因此在大多数情况下能够获得更优的轨迹。 [29]提出了可通行空间大凸区域的概念及其计算方法。在此基础上， [30] 提出了一种利用安全飞行走廊（SFC）概念，将轨迹生成构建为二次规划（QP）的方法。在自动驾驶领域， [31] 首次提出了时空语义走廊（SSC），以帮助车辆在复杂城市

环境中生成时空轨迹。Zhang等人提出了一种分层框架，包括粗略搜索、精细优化和基于安全带的碰撞规避 [32],，并利用时空走廊确保自车在时空地图上的轨迹安全高效。 [33] 将时空走廊定义为一系列具有时空连通性的体素，并通过选择相应的体素来实施行为规划。 [34]，[35]引入了梯形棱柱形走廊用于优化，以生成最终轨迹。然而，上述方法均使用贝塞尔曲线优化最终轨迹，这在一定程度上削弱了车辆模型约束，可能导致最终解不可行。目前，研究人员很少利用时空走廊在保障车辆安全和简化规划问题方面的特性来解决多车协同问题。

在多车辆协同运动规划中引入时空走廊具有诸多优势。时空走廊的统一表达增强了该框架的适用性，使所提方法不再局限于特定场景。此外，走廊内轨迹优化能够充分考虑车辆的运动学特性，从而提高安全性、质量和效率。

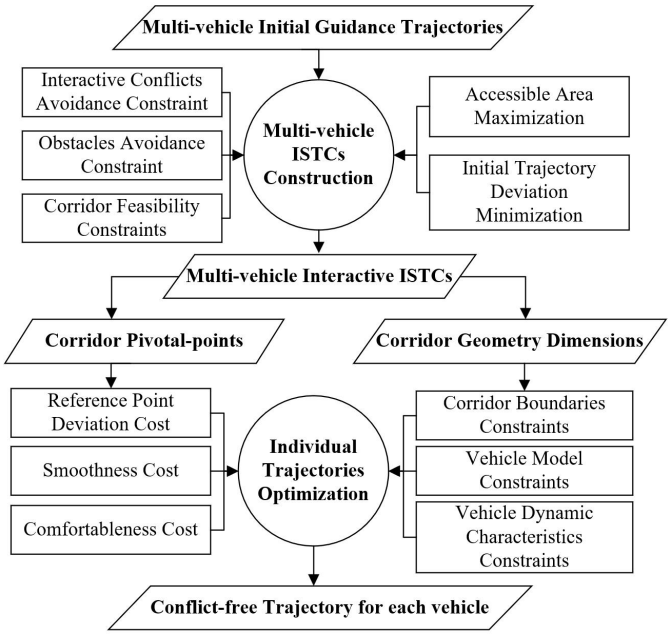


图 2. 算法结构示意图。

III. 框架

算法结构示意图如图2所示。整个分层方法由两个主要部分组成：多车辆 ISTC 构建和个体轨迹优化。第一部分中，中央计算机在获取所有车辆的意图信息和环境约束后，全局计算 ISTC，即为每辆车生成一条安全通道，使其穿过冲突区域。第二部分中，各车辆在各自的时空走廊内生成动态可行轨迹，无需再考虑交互冲突和障碍物碰撞。

在计算 ISTC 之前，初始引导轨迹作为车辆驾驶意图的输入，用于引导所有时空走廊的扩展。对于结构化城市环境，我们直接使用车道线作为引导；对于其他环境，则使用混合A*[36] 算法分别为每辆车寻找满足环境约束的路径。为路径赋予恒定速度以补充时间维度后，该路径便作为输入的初始引导轨迹。随后，采用混合整数线性规划（MIQP）构建 ISTC。在 MIQP 设计中，必须考虑两类冲突以确保安全：1）车辆之间的交互冲突；2）车辆与障碍物之间的环境冲突。此外，还需考虑另一类约束以确保可行性：3）走廊可行性约束。上述三类约束均可纳入 MIQP 设计，详见第 IV 节。至于目标函数，鉴于预期最大化每辆车的可达区域，并使走廊扩展与驾驶意图保持一致，我们将优化目标设定为：1）最大化每个时间单位内的走廊立方体；2）最小化走廊立方体关键点与初始引导轨迹之间的偏差。

在上述模块中，构建的 ISTC 已完全解决冲突，并在时空构型空间中为每辆车划定了各自的可通行区域。因此，整个多车运动规划问题可以解耦为多个单车轨迹优化问题。在时空走廊内生成最终轨迹的问题被转化为一个非线性规划（NLP）问题。轨迹优化需要在多个约束条件下考虑若干目标。值得注意的是，初始参考轨迹中的驾驶意图信息已传递给走廊中的关键点序列。因此，为确保轨迹符合车辆的意图，其中一个代价项是使生成的轨迹尽可能接近这些关键点。此外，还需考虑轨迹质量，因为舒适度代价和平滑性是需要评估的指标。另外，得益于 ISTC，原本用于生成无冲突轨迹的强非凸约束也被转化为线性走廊边界约束。最后，车辆模型约束和车辆动力学特性约束也被纳入非线性规划问题，以提高轨迹的动力学适用性。

在该框架中，构建 ISTC 旨在确保交互无冲突且无障碍物，轨迹优化则用于生成高质量的可执行轨迹。该方法不仅能妥善处理车辆之间的交互，还将多车运动规划解耦为单车轨迹优化。在此基础上，最终的轨迹优化可在各车辆的车载计算平台上完成，从而显著提升实时性能。

IV. 方法

A. 多车辆 *ISTC* 的构建

初始引导轨迹未考虑车辆之间的冲突，因此无法直接执行。然而，引导轨迹提供了每辆车的基本意图和行为信息，可用作划定各车辆无冲突可通行区域的参考。本节将详细介绍如何构建 ISTC 的混合整数二次规划（MIQP）。

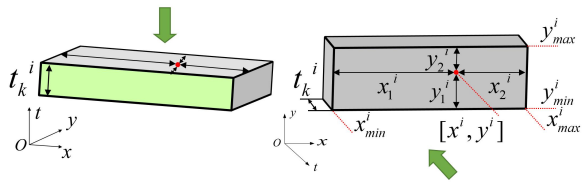


图3. 单个走廊立方体的结构。

由于车辆之间的冲突与其在每个时刻的位置有关，因此需要同时在时间域和空间域中考虑时空走廊。走廊的结构如图1所示，它表示车辆在一系列时间单位内可通行的区域。一个走廊立方体的具体形式如图3所示。车辆 i 在时间单位 t_k^i内的步长位置和大小由枢轴点 $p^i(k) = (x_k{}^i, y_k{}^i)$ 和尺度向量 $g^i(k) = [x_1^i(k)\ x_2^i(k)\ y_1^i(k)\ y_2^i(k)]$ 确定。因此，每个走廊立方体可表示为：

$$Cube^i(k) = [x_k{}^i, y_k{}^i, x_1^i(k), x_2^i(k), y_1^i(k), y_2^i(k)] \tag{1}$$

构建 ISTC 时主要考虑的两个性能指标如下：

$$J_{\text{area}}^i = \omega_{\text{area}}^i \sum_k \left(x_1^i(k) + x_2^i(k) + y_1^i(k) + y_2^i(k)\right) \tag{2}$$

$$J_{\text{ref}}^i = \omega_{\text{ref}}^i \sum_k \left(\left(x^i(k) - x_{\text{ref}}^i(k)\right)^2 + \left(y^i(k) - y_{\text{ref}}^i(k)\right)^2\right) \tag{3}$$

其中，$(x_{\text{ref}}^i(k)\ y_{\text{ref}}^i(k))$ 是初始参考轨迹提供的参考点，J_{area}^i 表示走廊立方体的宽度，J_{ref}^i 表示枢轴点与初始参考轨迹之间的偏差。

随后，可将混合整数二次规划模型表述如下：

$$\begin{aligned}
&\underset{p^i, g^i, \forall i \in \mathcal{V}}{\text{minimize}} \sum_i \eta^i \sum_k \left(-J_{\text{area}}^i\ (g^i(k)) + J_{\text{ref}}^i\ (p^i(k))\right) \\
&\text{s.t.} \\
&\quad \forall i \in \mathcal{V}, j \in \mathcal{V}\backslash\{i\}, k \in \mathcal{K},\ \text{obs}\ m \in \mathcal{O}: \\
&\quad Cube^i(k) \cap Cube^j(k) = \emptyset \\
&\quad Cube^i(k) \cap O^m(k) = \emptyset \\
&\quad Other\ \ feasibility\ \ constrains
\end{aligned} \tag{4}$$

其中，η^i 表示车辆的优先级权重，i表示车辆，$\mathcal{V}$ 表示车辆集合，$\mathcal{K}$ 表示时间步集合，$\mathcal{O}$ 表示障碍物集合，$Cube^i(k) \cap Cube^j(k) = \emptyset$ 表示用于避免车辆之间交互冲突的约束，而 $Cube^i(k) \cap O^m(k) = \emptyset$ 表示来自障碍物的环境约束。此外，为确保每条走廊对其车辆而言均具有动态可行性，以生成轨迹，本文还考虑了若干可行性约束。本文采用 Gurobi 求解器对模型进行求解，平均耗时可保持在毫秒级。

1）交互冲突规避约束：该约束旨在确保 ISTC 能够使车辆之间保持安全距离，也就是说，同一时间单位内不同车辆的走廊立方体不能相互重叠。图 4(a) 展示了同一时间单位内三辆车的走廊立方体作为示例。为确保互不干扰，需要考虑任意两个走廊立方体之间的位置关系。因此，该约束表述如下：

$$\begin{cases}
x_{min}^i(k) - x_{max}^j(k) + \delta_x^{ij}(k) \cdot M \geq \gamma_{x,v2v} \\
x_{min}^j(k) - x_{max}^i(k) + \delta_x^{ji}(k) \cdot M \geq \gamma_{x,v2v} \\
y_{min}^i(k) - y_{max}^j(k) + \delta_y^{ij}(k) \cdot M \geq \gamma_{y,v2v} \\
y_{min}^j(k) - y_{max}^i(k) + \delta_y^{ji}(k) \cdot M \geq \gamma_{y,v2v} \\
\delta_x^{ij}(k) + \delta_x^{ji}(k) + \delta_y^{ij}(k) + \delta_y^{ji}(k) \leq 3 \\
\delta_x^{ij}(k) \in \{0,1\}, \delta_x^{ji}(k) \in \{0,1\}, \\
\delta_y^{ij}(k) \in \{0,1\}, \delta_y^{ji}(k) \in \{0,1\}
\end{cases} \tag{5}$$

其中，M为最大值，$\delta_x^{ij}(k)$ $\delta_x^{ji}(k)$ $\delta_y^{ij}(k)$ $\delta_y^{ji}(k)$是取值为 1 或 0 的整数变量，γ_x、$v2v$ 和γ_y、$v2v$ 分别是车辆在 x 和 y 方向上的安全阈值，$[x_{min}^i(k)\ x_{max}^i(k)\ y_{min}^i(k)\ y_{max}^i(k)]$ 以及$[x_{min}^j(k)\ x_{max}^j(k)\ y_{min}^j(k)\ y_{max}^j(k)]$ 分别是同一时间单位内$Cube^i$ 和 $Cube^j$ 的边界。如果某个整数变量等于 1，则表示包含该整数变量的不等式恒成立，因此相应的约束被禁用。但如果整数变量等于 0，其作用如图 4(b) 所示。例如，当 $\delta_x^{ij}(k)$等于 0 时，只有当 $Cube^i$ 的左边界值 $x_{min}^i(k)$大于 $Cube^j$的右边界值 $x_{min}^i(k)$时，第一个不等式才成立。类似地，另外三个整数变量等于 0 时的情况也如图 4(b) 所示。四个整数变量中至少有一个等于 0，可确保在每个时间单位内，两辆车的可通行区域

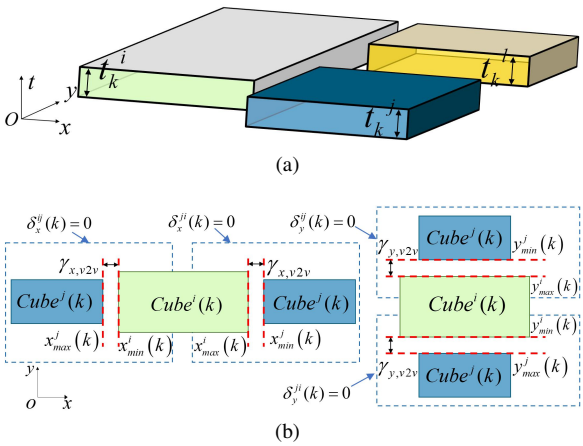


图 4. (a) 一个时间单位内三辆车的走廊立方体示例。(b) 施加相应约束后，走廊立方体在 $x-y$ 平面上的投影。

彼此独立。将此约束添加到所有车辆对中，即可确保整个交互过程的安全性。

2）障碍物避让约束：由于时空走廊能够为环境中的各种语义元素提供统一表示，我们也可以将障碍物表示为立方体框，并采用与（1）类似的方法确保车辆与障碍物之间不存在冲突，如图5所示。因此，约束可描述为：

$$\begin{cases} x_{min}^{i}(k)-x_{max}^{\text{obs }m}(k)+\delta_{x}^{i,\text{obs }m}(k)\cdot M\geq r_{x,v2o} \\ x_{min}^{\text{obs }m}(k)-x_{max}^{i}(k)+\delta_{x}^{\text{obs }m,i}(k)\cdot M\geq r_{x,v2o} \\ y_{min}^{i}(k)-y_{max}^{\text{obs }m}(k)+\delta_{y}^{i,\text{obs }m}(k)M\geq r_{y,v2o} \\ y_{min}^{\text{obs }m}(k)-y_{max}^{i}(k)+\delta_{y}^{\text{obs }m,i}(k)\cdot M\geq r_{y,v2o} \\ \delta_{x}^{i,\text{obs }m}(k)+\delta_{x}^{\text{obs }m,i}(k)+\delta_{y}^{i,\text{obs }m}(k)+\delta_{y}^{\text{obs }m,i}(k)\leq 3 \\ \delta_{x}^{i,\text{obs }m}(k)\in\{0,1\},\delta_{x}^{\text{obs }m,i}(k)\in\{0,1\}, \\ \delta_{y}^{i,\text{obs }m}(k)\in\{0,1\},\delta_{y}^{\text{obs }m,i}(k)\in\{0,1\} \end{cases} \tag{6}$$

其中，$[x_{\min}^{obs\ m}(k)\ x_{\max}^{obs\ m}(k)\ y_{\max}^{obs\ m}(k)\ y_{\max}^{obs\ m}(k)]$ 是时间单位 t_k^i 内障碍物 m 的立方体框，$r_{x,\ v2o}$ 和$r_{y,\ v2o}$是安全阈值，$\delta^{i,}{}_{x}^{\text{obs }m}(k)$、$\delta^{\text{obs }m,}{}_{x}{}^{i}(k)$、$\delta^{i,}{}_{y}{}^{\text{obs }m}(k)$、$\delta_{y}^{\text{obs }m\,i,}(k)$是整数变量。

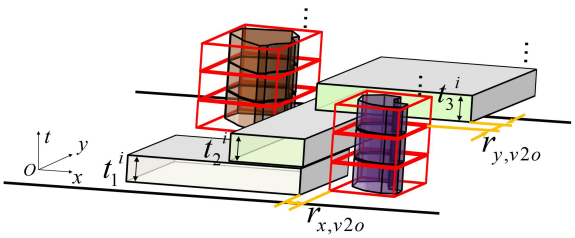


图5. 将障碍物表示为矩形的示例。

3）走廊可行性约束：确保每辆车的时空走廊与其他车辆及障碍物无冲突后，我们还需要添加约束，以确保走廊可供车辆执行，使其能在后续模块中生成可通行的轨迹。因此，构建 ISTC 时需要考虑车辆特性，并确保车辆能在指定时间到达每个走廊立方体。这类约束主要从两个方面考虑。一是每个时间单位内各个走廊立方体的尺寸和边界约束。每个走廊立方体都应能容纳车辆包围盒；此外，受车辆动力学特性的限制，每个走廊立方体的长度不得超过车辆在一个单位时间段内可行驶的最大距离。另一方面是连续时间段内相邻走廊立方体之间的尺寸和边界约束。考虑到车辆的行为连续性，相邻走廊立方体需要有最小重叠，以确保车辆能够从当前走廊立方体过渡到下一个走廊立方体。同时，相邻走廊立方体关键点之间的距离应根据车辆的最大速度加以限制。首先考虑一个时间单位内走廊立方体的尺寸和边界约束。每个走廊立方体都应有足够的空间，以容纳朝向不确定的车辆包围盒。

$$\begin{cases} x_1^i(k)+x_2^i(k)\geq\gamma_{x,car}^i \\ y_1^i(k)+y_2^i(k)\geq\gamma_{y,car}^i \end{cases} \tag{7}$$

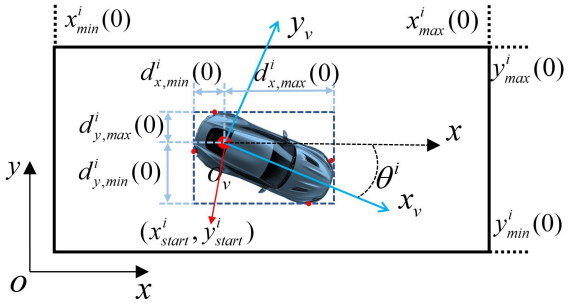


图 6. 初始时刻车辆包围盒和走廊立方体在 $x-y$ 平面上的投影

其中，$\gamma_{x、\ car}^i$、$\gamma_{y、\ car}^i$ 是每个时间段内容纳车辆几何形状所需的沿 x 和 y 轴方向的最小距离阈值。实际应用中，我们设定 $\gamma_{x,\ car}^i=\gamma_{y,\ car}^i=\sqrt{(l^i)^2+(w^i)^2}$，，其中 l^i 和 w^i 分别为车辆的长度和宽度。此外，由于车辆的初始状态是确定且不可调整的，相应的走廊立方体受到更严格的约束，如图6所示。应用以下约束：

$$\begin{cases} x_{min}^i(0)\leq x_{start}^i+d_{x,min}(0) \\ x_{max}^i(0)\geq x_{start}^i+d_{x,max}(0) \\ y_{min}^i(0)\leq y_{start}^i+d_{y,min}(0) \\ y_{max}^i(0)\geq y_{start}^i+d_{y,max}(0) \end{cases} \tag{8}$$

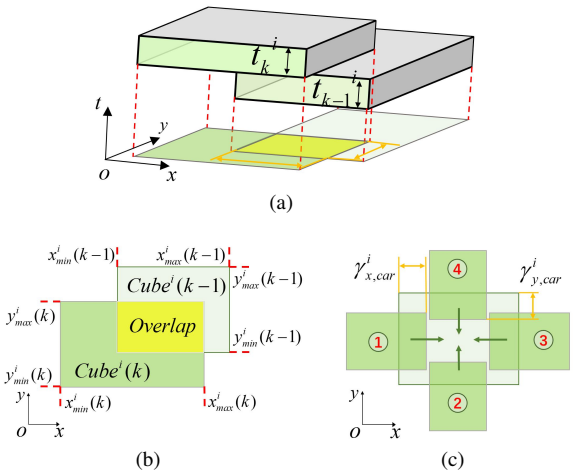
其中，$(x_{start}^i,\ y_{start}^i)$ 是车辆的初始位置，$[x_{min}^i(0)\ x_{max}^i(0)\ y_{min}^i(0)\ y_{max}^i(0)]$ 是起始走廊立方体在 x 和 y 方向上的边界，$d_{x,\ min}(0)$、$d_{x,\ max}(0)$、$d_{y,\ min}(0)$、$d_{y,\ max}(0)$ 是车辆包围盒边缘与车辆位置之间的距离，可根据车辆的状态和几何形状计算得到。此外，由于车辆的最大速度和加速度受到限制，单位时间段内每个走廊立方体的尺寸也受到限制。车辆 i 在时间单位 t_k^i 内的行驶范围约束如下：

$$\begin{cases} x_{min}^i(k)\geq\alpha_x^i\cdot x_{min,drive}^i(k) \\ x_{max}(k)\leq\alpha_x^i\cdot x_{max,drive}^i(k) \\ y_{min}^i(k)\geq\alpha_y^i\cdot y_{min,drive}^i(k) \\ y_{max}^i(k)\leq\alpha_y^i\cdot y_{max,drive}^i(k) \end{cases} \tag{9}$$

其中，$x_{min,drive}^i(k)$、$x_{max,drive}^i(k)$、$y_{min,drive}^i(k)$、$y_{max,\ drive}^i(k)$分别是车辆在 x 和 y 方向上的行驶范围最小边界和最大边界。此外，请注意，设置松弛因子 α_x^i,α_y^i 是为了提高求解混合整数二次规划问题的成功率。通常，我们将 α_x^i 和 α_y^i 设为 2。四个行驶范围值可通过以下公式计算：

$$
\begin{aligned}
&x^i_{min,drive}(k) = \\
&\quad \begin{cases} x^i(k-1) + \gamma^i_{s-} \cos\theta^i_{\text{ref}}(k-1), & \text{if } \cos\theta^i(k-1) \geq 0 \\ x^i(k-1) + \gamma^i_{s+} \cos\theta^i_{\text{ref}}(k-1), & \text{else} \end{cases} \\
&x^i_{max,drive}(k) = \\
&\quad \begin{cases} x^i(k-1) + \gamma^i_{s-} \cos\theta^i_{\text{ref}}(k-1), & \text{if } \cos\theta^i(k-1) \leq 0 \\ x^i(k-1) + \gamma^i_{s+} \cos\theta^i_{\text{ref}}(k-1), & \text{else} \end{cases} \\
&y^i_{min,drive}(k) = \\
&\quad \begin{cases} y^i(k-1) + \gamma^i_{s-} \sin\theta^i_{\text{ref}}(k-1), & \text{if } \sin\theta^i(k-1) \geq 0 \\ y^i(k-1) + \gamma^i_{s+} \sin\theta^i_{\text{ref}}(k-1), & \text{else} \end{cases} \\
&y^i_{max,drive}(k) = \\
&\quad \begin{cases} y^i(k-1) + \gamma^i_{s-} \sin\theta^i_{\text{ref}}(k-1), & \text{if } \sin\theta^i(k-1) \leq 0 \\ y^i(k-1) + \gamma^i_{s+} \sin\theta^i_{\text{ref}}(k-1), & \text{else} \end{cases}
\end{aligned} \tag{10}
$$

其中，γ^i_{s+}、γ^i_{s-} 是车辆 i 沿当前航向角向前和向后行驶的最大距离阈值，$\theta^i(k-1)$是根据初始引导轨迹获得的时间单位 t^i_{k-1} 处的航向角。维度



(a) (b) (c)

图7。（a）相邻走廊立方体之间的重叠约束；（b）两个相邻走廊立方体重叠部分的二维 $x-y$ 平面投影；（c）相邻走廊立方体重叠的四种基本情况。

还需要考虑连续时间段内相邻走廊立方体的边界约束。为了确保车辆能够在连续的走廊立方体之间可行地过渡，应对两个连续走廊立方体施加最小重叠要求，如图 5(a) 所示。在图 5(b) 所示的 x-y 平面投影中，我们可以更直观地观察两个走廊立方体之间的重叠关系。只要分别沿 x 和 y 方向约束位置关系，就能保证两个走廊立方体之间有足够的重叠面积。图 5(c) 展示了四种基本的重叠情形，箭头指向嵌套面积增大的方向。其他满足要求的重叠情形均可由这四种基本情形演变而来。约束可描述如下：

$$
\begin{cases} x^i_{max}(k) - x^i_{min}(k-1) \geq \gamma^i_{x,car} \\ x^i_{min}(k) - x^i_{max}(k-1) \leq \gamma^i_{x,car} \\ y_{max}(k) - y_{min}(k-1) \geq \gamma^i_{y,car} \\ y^i_{min}(k) - y^i_{max}(k-1) \leq \gamma^i_{y,car} \end{cases} \tag{11}
$$

其中 $\gamma^i_{x,car}$,$\gamma^i_{y,car}$ 的含义与公式 (7) 中相同。最后，连续走廊立方体关键点之间的行驶距离受车辆最大速度的限制，可描述如下：

$$
\begin{cases} |x^i(k) - x^i(k-1)| \leq |\gamma^i_{s+} \cos\theta_{ref}(k-1)| \\ |y^i(k) - y^i(k-1)| \leq |\gamma_{s+} \sin\theta^i_{ref}(k-1)| \end{cases} \tag{12}
$$

其中，$|x^i(k)-x^i(k-1)|$ 和 $|y^i(k)-y^i(k-1)|$ 分别表示 x 和 y 方向上两个相邻关键点之间的距离；该公式将它们限制为小于 x 轴和 y 轴上最大前向距离的分量。

B. 多车辆无冲突轨迹生成

构建ISTC后，整个多车辆规划问题被解耦为单车辆轨迹规划问题。每辆车的时空走廊有两个特征对轨迹优化起着重要作用，即关键点和走廊边界。前者用于引导最优轨迹的方向，后者则可将轨迹限制在安全且可行的行驶区域内。由于每条走廊都是在三维时空构型空间中构建的，因此我们可以采用横纵向耦合方法求解轨迹优化问题，并直接在时空地图中生成轨迹。

在本模块中，将考虑三项代价以获得最终轨迹：平滑度代价 J^i_{smooth}、舒适度代价 $J^i_{comfort}$ 以及偏离关键点代价 $J^i_{pivotal}$。平滑度代价与轨迹的曲率 κ^i_t 有关：

$$
J^i_{smooth} = \omega^i_\kappa \sum_t \kappa^{i2}_t \tag{13}
$$

其中，ω^i_κ 是车辆 i的平滑度权重。舒适度代价由纵向部分和横向部分组成：

$$
J^i_{comfort} = \omega^i_\beta \sum_t \left(\beta^i_t\right)^2 + \omega^i_j \sum_t \left(j^i_t\right)^2 \tag{14}
$$

其中，β^i_t 是前轮转角变化率，其权重为 ω^i_β；j^i_t 是加速度变化率，其权重为 ω^i_j。偏离关键点代价如下：

$$
\begin{aligned}
J^i_{pivotal} = \omega^i_{px} &\sum_{k,t\in[t_k,t^i_{k+1})} \left(x^i_t - x^i(k)\right)^2 \\
&+ \omega^i_{py} \sum_{k,t\in[\in t_k,t^i_{k+1})} \left(y^i_t - y^i(k)\right)^2
\end{aligned} \tag{15}
$$

其中，ω^i_{px} 和ω^i_{py} 分别表示车辆 i 在 x 和 y 方向上相对于枢轴点的偏差权重。

然后，可将非线性规划问题表述为：

$$\begin{aligned}
&\underset{\beta_t^i, j_t^i}{\text{minimize}} \ \left(J_{smooth}^i + J_{comfort}^i + J_{pivotal}^i\right) \\
&s.t. \\
&\quad \forall vehicle\ i, time\ t, \beta_t^i \in \mathbb{R}, j_t^i \in \mathbb{R} \\
&\quad S_{t+1}^i = f(S_t^i, \beta_t^i, j_t^i) \\
&\quad \sum_{t=0}^{t} \beta_t^i \in [-\delta_{\max}^i, \delta_{\max}^i], \sum_{t=0}^{t} j_t^i \in [-a_{\text{dec max}}^i, a_{\text{acc max}}^i] \\
&\quad (x_t^i, y_t^i) \in Corridor^i
\end{aligned} \tag{16}$$

其中，$[-\delta_{\max}^i, \delta_{\max}^i]$ 是前轮转向角的范围，$[-a_{\text{dec max}}^i, a_{\text{acc 最大值}}^i]$ 是加速度的范围，S_t^i是车辆 i 在时刻 t 的状态参数集，$S_{t+1}^i = f(S_t^i, \beta_t^i, j_t^i)$是在每个时间步满足车辆模型等式约束的状态转移，可表示如下：

$$\begin{bmatrix} x_{t+1}^i \\ y_{t+1}^i \\ \theta_{t+1}^i \\ \delta_{t+1}^i \\ v_{t+1}^i \\ a_{t+1}^i \end{bmatrix} = \begin{bmatrix} x_t^i + v_t^i \cos\theta_t^i \Delta t \\ y_t^i + v_t^i \sin\theta_t^i \Delta t \\ \theta_t^i + v_t^i \frac{\tan\delta_t^i}{L} \Delta t \\ \delta_t^i \\ v_t^i + a_t^i \Delta t \\ a_t^i \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \Delta t & 0 \\ 0 & 0 \\ 0 & \Delta t \end{bmatrix} \begin{bmatrix} \beta_t^i \\ j_t^i \end{bmatrix} \tag{17}$$

上述方程是轮式车辆的离散运动学自行车模型，假设车辆行驶在相对平坦的道路上，且不考虑垂向运动和侧倾运动。车辆的动力学特性约束，例如最大加速度和最大前轮转向角，描述如下：

$$\begin{cases} -\delta_{max}^i \le \delta_t^i \le \delta_{max}^i \\ -a_{\text{dec } max}^i \le a_t^i \le a_{\text{acc } max}^i \end{cases} \tag{18}$$

此外，$(x_t^i, y_t^i) \in Corridor^i$ 是用于使轨迹保持在时空走廊内的约束，这意味着车辆的包围盒在任何时刻都应由其时空走廊的立方体包围，类似于图6所示的起始位置，该约束可描述为：

$$\begin{cases} x_{min}^i(k) \le x_{t,rr}^i \le x_{max}^i(k), y_{min}^i(k) \le y_{t,rr}^i \le y_{max}^i(k) \\ x_{min}^i(k) \le x_{t,fl}^i \le x_{max}^i(k), y_{min}^i(k) \le y_{t,fl}^i \le y_{max}^i(k) \\ x_{min}^i(k) \le x_{t,rl}^i \le x_{max}^i(k), y_{min}^i(k) \le y_{t,rl}^i \le y_{max}^i(k) \\ x_{min}^i(k) \le x_{t,rr}^i \le x_{max}^i(k), y_{min}^i(k) \le y_{t,rr}^i \le y_{max}^i(k) \end{cases} \tag{19}$$

V. 实验结果与讨论

本节安排了若干组无信号交叉口场景及具有挑战性的密集场景仿真实验。在所有实验中，我们以秒为单位对时间进行离散化，以求解用于构建 ISTC 的混合整数二次规划问题；也就是说，每个走廊立方体代表一秒内的可通行区域。

所提算法采用 C++ 实现，并在搭载 Ubuntu 18.04、配备 Intel i7-8700H @2.30 GHz 中央处理器和 16 GB 随机存取存储器的笔记本电脑上运行。我们使用 lpsolve求解器和 Gurobi 求解器求解构建 ISTC 的混合整数二次规划，并使用 IPOPT 求解非线性规划以生成各时空走廊内的最终轨迹。此外，我们使用 GridMap 表示环境的占用栅格地图。

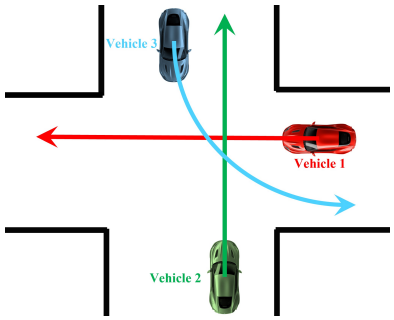


图8. 三辆车在无信号交叉路口进行仿真。

A. 无信号交叉路口仿真实验

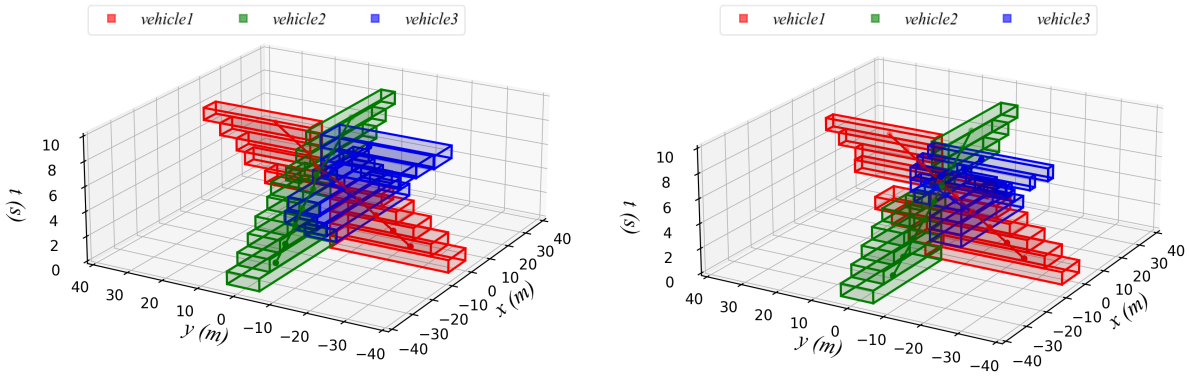
1）描述：图8展示了无信号交叉路口场景中三辆来自不同方向的车辆的仿真。 $vehicle1$ 和 $vehicle2$ 来自不同方向，意图直行； $vehicle3$ 来自与前两辆车不同的方向，意图左转。在这一结构化场景中，可根据车辆的行驶意图并结合车道线来投影初始引导轨迹。假设所有车辆均以 $5m/s$的匀速通行。随后，图8中用不同颜色的线表示每辆车的初始引导轨迹。假设所有车辆的尺寸参数为：长度 = $4m$、宽度 = $2m$、轴距 = $2.7m$，道路宽度为 $10m$。此外，算法的相关参数设置见表I。为比较不同优先级权重的影响，进行了两组实验。第一组中，左转的 $vehicle3$ 具有较高优先级；第二组中，则赋予直行的 $vehicle1$ 和 $vehicle2$较高优先级，如表 II 所示。在两组实验中，最小优先级权重均近似为0，以增大目标函数的梯度并提高求解速度。

表I 所提方法的相关参数

参数	值	参数	值	参数	值
$\gamma_{s+}^{1,\ 2,\ 3}$	10	$w_{ref}^{1,2,3}$	10.00	$w_j^{1,2,3}$	1.0
$\gamma_{s-}^{1,2,3}$	5	$w_{\kappa}^{1,2,3}$	1.0	$w_{,px}^{1\ 2\ 3}$	1.0
$w_{,area}^{1\ 2\ 3}$	0.50	$w_{\beta}^{1,2,3}$	100.0	$w_{py}^{1,2,3}$	1.0

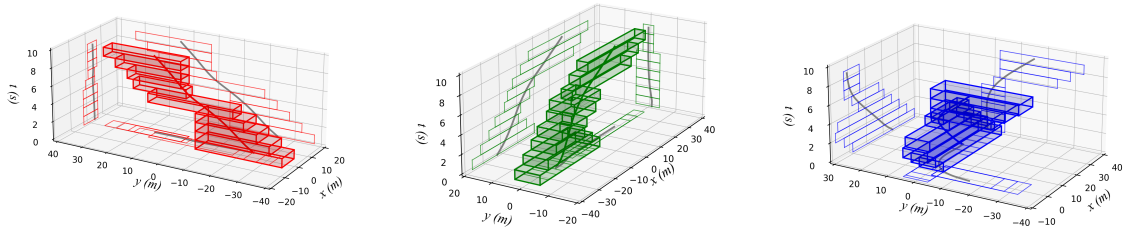
表 II 转弯优先场景和直行优先场景中各车辆的优先级权重

组	η^1	η^2	η^3
I	0.20	0.01	0.50
II	0.50	0.40	0.01

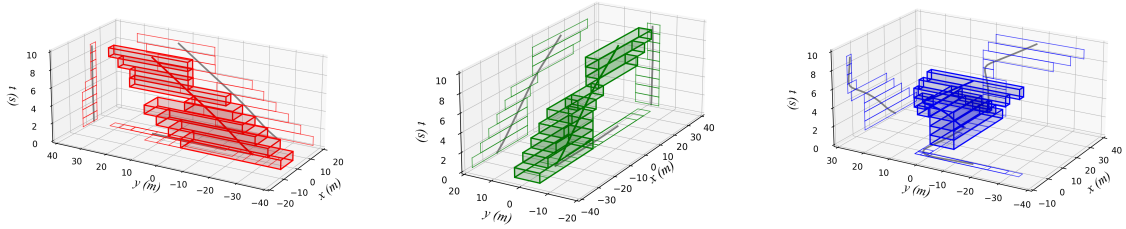


(c) 第 I 组的 ISTC 结果　　(d) 第 II 组的 ISTC 结果

图9。 对比结　　果：多车辆 ISTC 的对比结果。（每个时空走廊内均包含关键点及其连接线　　s)



(a) 车辆1，$\eta^1 = 0.20$　(b) 车辆2，$\eta^2 = 0.01$　(c) 车辆3，$\eta^3 = 0.50$



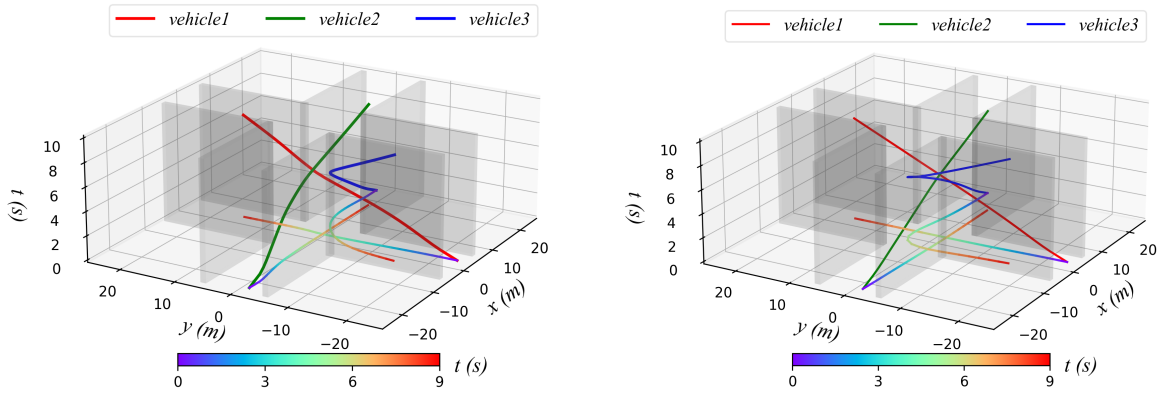
(d) 车辆1，$\eta^1 = 0.50$　(e) 车辆2，$\eta^2 = 0.40$　(f) 车辆3，$\eta^3 = 0.01$

图10. 个体时空走廊及其内部的最终轨迹结果；(a)(b)(c)为第 I 组的结果，(d)(e)(f)为第 II 组的结果。

2）结果与讨论：两组构建的ISTC均按预期生成，如图9所示。根据初始引导轨迹，每辆车的走廊立方体沿正确方向扩展，直至驶离冲突区域。此外，不同车辆的时空走廊在任何时间单位内均不重叠，从而确保冲突消解过程中的安全性。

比较 ISTC 的结果可以看出，不同优先级权重的影响截然不同。优先级权重较高的车辆倾向于生成面积更大的时空走廊，且走廊立方体的关键点往往更靠近初始参考轨迹。相反，优先级权重较低的车辆更倾向于牺牲自身的行驶空间，为其他高优先级车辆让出空间。不过，无论优先级权重较高还是较低，由于采用了第 IV 节中设定的约束，每辆车的走廊始终安全且满足动力学要求。在第 I 组中，$vehicle3$ 的时空走廊在冲突开始时占据交叉路口中心的倾向最为明显。而在第 II 组中，交叉路口中心大部分被 $vehicle1$的走廊占据。此外，图 10 展示了各车辆单独的时空走廊及其走廊内轨迹结果，这也体现了优先级权重的影响。比较图 10(c) 和图 10(f) 中左转车辆$vehicle3$ 的走廊和轨迹可知，第 I 组的结果明显优于第 II 组，主要体现在两个方面：1）走廊占据的总体空间更大。2）生成的最终轨迹更加平滑。相比之下，在包含 $\eta^3 = 0.01$的直行优先组中，$vehicle3$ 的最终轨迹受到狭窄走廊空间的严格限制。

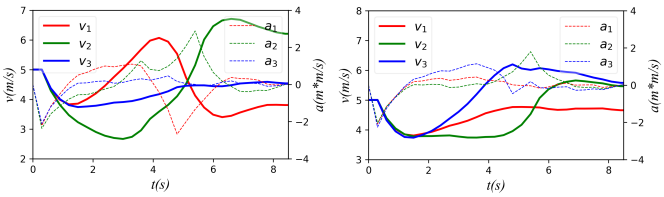
多车辆轨迹结果以及速度和加速度结果体现了所提方法在 3-D(x，y，t) 时空构型空间中的灵活性，分别如图 11 和图 12 所示。在第 I 组中，$vehicle3$ 在初始引导下以最高优先级权重进行了一次舒适的大半径左转。同时，根据图 12(a)，$vehicle1$ 和 $vehicle2$ 主要通过调整各自的速度向 $vehicle3$ 让行。相比之下，在第 II 组中，左转的 $vehicle3$ 选择绕行核心冲突区域，并以较小半径转弯，舒适性较低。而



(a) 第 I 组轨迹结果 (b) 第 II 组轨迹结果

图1 1. 最终多车辆轨迹的比较结果。

$vehicle1$ 和 $vehicle2$ 则因此受益，获得了更加稳定的速度，如图 12(a) 所示。此外，图 12 中的两组实验均表明，所有车辆在开始进入冲突区域时都选择减速。速度和加速度的波动幅度与车辆优先级权重的相对大小呈负相关。



(a) 组 I (b) 组 II

图12. 无信号交叉路口中的速度和加速度结果

表 III 所示的耗时体现了所提算法的效率。由于分层框架能够通过 ISTC 将多车辆协同运动规划解耦为单车轨迹优化，因此耗时可分为两部分。在两组实验中，框架的第一层都能够快速构建 ISTC。在第二层中，每辆车都能在充分考虑车辆运动学特性的前提下，以可接受的耗时安全、高质量地通过冲突区域。

表 III 整个方法的耗时

组	t_{layer1} (s)	t_{layer2}(秒)		
		$vehicle1$	$vehicle2$	$vehicle3$
I	0.092	0.146	0.042	0.167
II	0.073	0.074	0.053	0.246

总之，无信号交叉口区域的实验结果表明，该方法符合预期，与第四节所述方法的初衷一致。ISTC 能够安全、高效地解决交互冲突，并简化协作轨迹生成问题。灵活的参数设置使车辆在相同环境和初始条件下通过冲突区域时能够采取不同策略。此外，在三维时空构型空间中寻找解，大大提高了车辆避让冲突的灵活性。

B. 复杂密集场景下与基线算法的比较

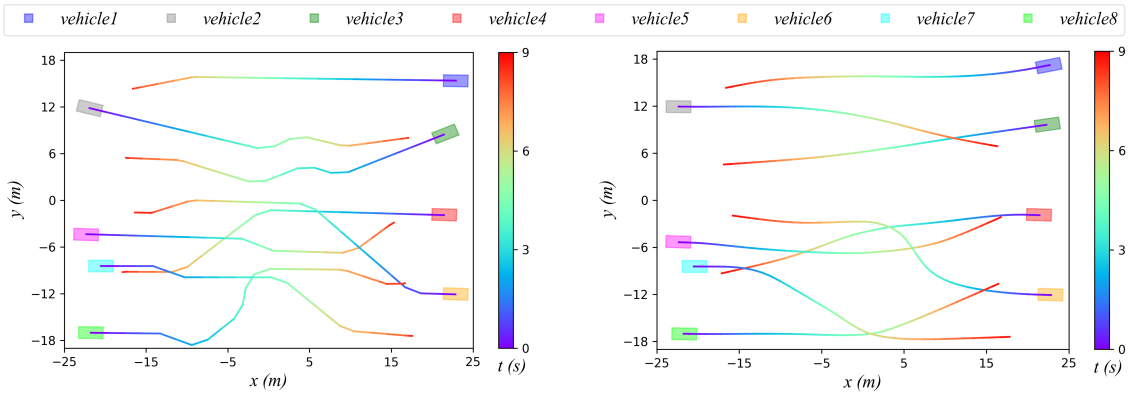
1）描述：为验证所提分层方法在一般场景中的可行性和优势，我们将其与[12]中提出的CL-CBS进行了对比实验。实验在一张50m*38m的地图中进行，分别测试了4辆、6辆和8辆车辆在地图中部有障碍物和无障碍物两种情况下的表现。为了增加冲突解决难度，车辆的初始位置设在地图左侧或右侧，并要求所有车辆无碰撞地驶向另一侧。在我们的方法中，初始引导轨迹通过混合A*求得，并将恒定速度设为5m/s。两种方法的仿真均采用相同的起点状态和目标状态。

对比结果如表IV所示。对于CL-CBS，N_{node} 表示冲突基搜索扩展的冲突节点数，t_{total} 表示完成多车辆规划任务的总耗时，L_{total} 表示所有车辆轨迹的总长度。对于所提方法，t_{l_1} 表示第一层构建ISTC的耗时，t_{l_2} 表示第二层优化所有车辆轨迹的耗时，t_{total} 和L_{total} 的含义与CL-CBS中的相同。此外，图13列出了8辆车辆在有障碍物和无障碍物情况下的行驶轨迹。

2）结果与讨论：两种方法的耗时通常都会随着障碍物和车辆数量的增加而增加，但我们的方法对这些因素不太敏感，在更密集的场景中表现更好。在所提出的框架中，MIQP 的第一层能够

表IV 复杂密集场景下与基线算法的比较结果。

车辆数字	障碍物	CL-CBS			所提方法			
		N_{node}	$t_{total}(s)$	$L_{total}(m)$	$t_{l_1}(s)$	$t_{l_2}(s)$	$t_{total}(s)$	$L_{total}(m)$
4	✗	5	0.069	175.554	0.171	0.402	0.573	157.615
4	✓	3	0.065	182.119	0.215	0.415	0.630	159.735
6	✗	6	0.228	258.018	0.651	0.793	1.444	235.376
6	✓	225	9.611	272.611	1.622	0.774	2.436	239.282
8	✗	530	22.98	352.066	3.954	0.977	4.931	313.634
8	✓	737	38.090	381.616	6.369	1.002	7.371	319.149



(a) CL-CBS，无障碍物 (b) 所提方法，无障碍物

(c) CL-CBS，有障碍物 (d) 所提方法，有障碍物

图13. 复杂密集场景下与基线算法的比较结果

在所有仿真中高效地为所有参与车辆构建 ISTC。ISTC 将多车辆协作问题解耦，因此第二层针对每辆车单独求解时，其计算复杂度不受整体问题复杂度增加的影响，因而第二层的总耗时与车辆数量呈正线性关系。相比之下，在充满冲突的密集场景中，CL-CBS 的冲突节点数量激增，导致耗时急剧上升。这一现象的原因在于，CL-CBS 在底层规划器中并未直接考虑车辆之间的协同，而是先为每辆车独立规划轨迹，再通过检测冲突位置扩展冲突节点。对于每个冲突节点，CL-CBS 也不会同时协调所有涉及冲突的车辆来解决冲突，而是每次仅对一辆车施加约束，使其绕开冲突位置。

此外，在所有仿真中，我们的方法都能让车辆以相对更短的路径通过冲突区域，因为 ISTC 的构建和轨迹优化可以在三维 (x，y，t) 构型空间中调整车辆的位置和速度，从而保留更大的解空间。图 13 中的结果还表明，我们的方法能够生成更精细、平滑的轨迹。如图 13(d) 所示，ISTC 可以充分利用更大的解空间，在确保车辆安全的同时提高通行效率。例如，$vehicle2$ 和 $vehicle3$ 的轨迹在狭窄开口处相遇，并列通过；而 $vehicle5$ 的轨迹则迅速通过最近的通道，为 $vehicle4$ 和 $vehicle6$ 腾出空间。相比之下，如图 13(c) 所示，CL-CBS 生成的轨迹过于离散，甚至导致车辆越过边界或与障碍物碰撞，例如

$vehicle2$ 和 $vehicle3$ 的轨迹。

然而，基于冲突基搜索的 CL-CBS 在自由空间充足时寻找可行轨迹的效率极高，速度远快于本文提出的基于ISTC的方法。出现这一现象的原因是，CL-CBS 的冲突解决机制是在时空约束下迭代地重新规划单个冲突车辆的轨迹。在车辆较少、自由空间较大的场景中，这种机制在解决当前冲突后引发新冲突的概率较低，从而大幅减少了冲突节点扩展和混合A*重新规划的次数。相比之下，我们的方法采用混合整数二次规划和非线性规划进行双层优化，同时对所有车辆进行全局协同，因此生成更平滑轨迹所需的时间更长。

VI. 结论

本研究提出了一种新颖的分层框架，用于在一般场景中实现多车辆协同运动规划。第一层设计的混合整数二次规划能够高效构建ISTC，为复杂交互环境中的每辆车辆开辟安全的时空走廊。混合整数二次规划施加的约束能够确保车辆安全，同时为可行轨迹生成保留充足余量。基于此，第二层采用的非线性规划能够考虑非完整运动学，有效生成安全、平滑且位于走廊内的轨迹。得益于整个框架在三维（x，y，t）时空构型空间中进行优化，它保留了更大的解空间，并提升了规划结果的质量。与其他现有方法相比，所提出的双层方法能够将全局多车辆运动规划任务解耦为各个个体的任务，从而简化问题复杂度并分散计算压力。

在未来的工作中，我们将尝试设计比立方体更灵活的时空走廊结构，以进一步提高算法的适应性和鲁棒性。

参考文献

[1] J. E. Siegel、D. C. Erb 和 S. E. Sarma，“网联车辆领域综述——架构、赋能技术、应用及发展方向”，*IEEE*智能交通系统汇刊，第19卷，第8期，第2391–2406页，2017年。[2] J. Rios-Torres 和 A. A. Malikopoulos，“网联自动驾驶车辆在交叉路口和高速公路匝道入口汇入时协调的综述”，*IEEE*智能交通系统汇刊，第18卷，第5期，第1066–1077页，2016年。[3] J. Yu 和 S. M. LaValle，“图上的最优多机器人路径规划的结构与难解性”，载于第二十七届*AAAI*人工智能会议，2013年。[4] D. Silver，“协作式寻路”，载于*AAAI*人工智能与交互式数字娱乐会议论文集，第1卷，第1期，2005年，第117–122页。[5] M. Barer、G. Sharon、R. Stern 和 A. Felner，“多智能体寻路问题中基于冲突的搜索算法的次优变体”，载于第七届组合搜索年会，2014年。[6] D. Le 和 E. Plaku，“通过多智能体搜索引导的协同采样扩展进行动态环境下的多机器人运动规划”，*IEEE*机器人与自动化快报，第4卷，1875，2019年2，第1868页。[7] M. M. Khorshid、R. C. Holte 和 N. R. Sturtevant，“用于非最优多智能体寻路的多项式时间算法”，载于第四届组合搜索年会，2011年。

[8] G. Wagner 和 H. Choset，“M*：一种具有性能界限的完整多机器人路径规划算法”，发表于*2011*年*IEEE/RSJ*智能机器人与系统国际会议。IEEE，2011年，第3260–3267页。[9] T. Standley，“为协作式寻路问题寻找最优解”，发表于*AAAI*人工智能会议论文集，第24卷，第1期，2010年，第173–178页。[10] M. Phillips 和 M. Likhachev，“SIPP：动态环境下的安全间隔路径规划”，发表于*2011*年*IEEE*机器人与自动化国际会议。IEEE，2011年，第5628–5635页。[11] A. Andreychuk、K. Yakovlev、E. Boyarski 和 R. Stern，“改进连续时间冲突基搜索”，发表于*AAAI*人工智能会议论文集，第35卷，第13期，2021年，第11 220–11 227页。[12] L. Wen、Y. Liu 和 H. Li，“CL-MAPF：面向具有运动学和时空约束的类汽车机器人的多智能体路径规划”，《机器人与自主系统》，第150卷，文章编号103997，2022年。[13] J. Li、M. Ran 和 L. Xie，“通过优先级轨迹优化实现多台非完整移动机器人的高效轨迹规划”，《IEEE机器人与自动化快报》，第6卷，第2期，第405–412页，2020年。[14] R. Reijnen、Y. Zhang、W. Nuijten、C. Senaras 和 M. Goldak-Altgassen，“结合深度强化学习与搜索启发式求解基于路段布局的多智能体路径规划”，发表于*2020*年*IEEE*计算智能研讨会系列（*SSCI*）。IEEE，2020年，第2647–2654页。[15] M. Sinkar、M. Izhan、S. Nimkar 和 S. Kurhade，“使用动态分布式深度学习模型进行多智能体路径规划”，发表于*2021*年通信、信息与计算技术国际会议（*ICCICT*）。IEEE，2021年，第1–6页。[16] C. Bai、P. Yan、W. Pan 和 J. Guo，“基于学习的多机器人编队控制与避障”，《IEEE智能交通系统汇刊》，第23卷，第8期，第11 811–11 822页，2021年。[17] M. Cai、Q. Xu、C. Chen、J. Wang、K. Li、J. Wang 和 X. Wu，“基于多车辆编队控制的灵活车道方向多车道无信号交叉口协作”，《IEEE车辆技术汇刊》，2022年。[18] W. Wang、C. Zhang、P. Wang 和 C.-Y. Chan，“利用半随机势场学习多车辆时空交互的表征”，发表于*2020*年*IEEE*智能车辆研讨会（*IV*）。IEEE，2020年，第1935–1940页。[19] C. Xia、M. Xing 和 S. He，“无交通标志交叉口场景下的自动驾驶交互式规划”，《IEEE智能交通系统汇刊》，第23卷，2022年12月，第24 818–24 828页。[20] Z. Liang、J. Cao、S. Jiang、D. Saxena 和 H. Xu，“结合对手建模的分层强化学习，用于分布式多智能体协作”，发表于*2022*年*IEEE*第*42*届分布式计算系统国际会议（*ICDCS*）。IEEE，2022年，第884–894页。[21] B. Li、Y. Zhang、Y. Feng、Y. Zhang、Y. Ge 和 Z. Shao，“平衡计算速度与质量：一种用于网联自动驾驶车辆协作式变道的分散式运动规划方法”，《IEEE智能车辆汇刊》，第3卷，2018年3月，第340–350页。[22] S. Xie、J. Hu、P. Bhowmick、Z. Ding 和 F. Arvin，“通过人工势场实现安全自动驾驶车辆超车的分布式运动规划”，《IEEE智能交通系统汇刊》，第23卷，第11期，第21 531–21 547页，2022年。[23] R. Deng、B. Di 和 L. Song，“基于蜂窝车联网的自动驾驶中超车机动的协作式碰撞规避”，《IEEE车辆技术汇刊》，第68卷，2019年5月，第4434–4446页。[24] Z. e. a. Kherroubi、S. Aknine 和 R. Bacha，“网联自动驾驶车辆高速公路入口匝道汇入的新型决策策略”，《IEEE智能交通系统汇刊》，第23卷，第8期，第12 490–12 502页，2022年。[25] H. Liu、W. Zhuang、G. Yin、R. Li、C. Liu 和 S. Zhou，“混合交通中基于控制障碍函数的网联自动驾驶车辆分散式入口匝道汇入控制”，发表于*2021*年*IEEE*智能交通系统国际会议（*ITSC*），2021年，第1125–1131页。[26] A. Nakamura、Y.-C. Liu 和 B. Kim，“用于碰撞规避的短期多车辆轨迹规划”，《IEEE车辆技术汇刊》，第69卷，第9期，第9253–9264页，2020年。[27] B. Wang、Y. Lu、T. Zhang、J. Gong 和 H. Zhao，“基于交互基元树的多车辆协同运动规划”，发表于*2021*年*IEEE*无人系统国际会议（*ICUS*）。IEEE，2021年，第51–57页。

[28] T. Kessler 和 A. Knoll，“面向自动驾驶的协作式多车辆行为协调”，发表于*2019*年*IEEE*智能车辆研讨会（*IV*）。IEEE，2019年，第1953–1960页。[29] R. Deits 和 R. Tedrake，“通过半定规划计算无障碍物空间的大型凸区域”，发表于《机器人算法基础*XI*：第十一届机器人算法基础国际研讨会精选论文》。Springer，2015年，第109–124页。[30] S. Liu、M. Watterson、K. Mohta、K. Sun、S. Bhattacharya、C. J. Taylor 和 V. Kumar，“在三维复杂环境中利用安全飞行走廊规划四旋翼飞行器的动态可行轨迹”，《IEEE机器人与自动化快报》，第2卷，第3期，第1688–1695页，2017年。[31] W. Ding、L. Zhang、J. Chen 和 S. Shen，“利用时空语义走廊为复杂城市环境生成安全轨迹”，《IEEE机器人与自动化快报》，第4卷，2019年3月，第2997–3004页。[32] T. Zhang、M. Fu、W. Song、Y. Yang 和 M. Wang，“基于时空地图并由安全带保证碰撞规避的轨迹规划”，《IEEE智能交通系统汇刊》，2020年。[33] T. Zhang、W. Song、M. Fu、Y. Yang、X. Tian 和 M. Wang，“一种基于时空体素、集成决策与轨迹规划的统一框架，用于高速公路自动驾驶”，《IEEE智能交通系统汇刊》，2021年。[34] S. Deolasee、Q. Lin、J. Li 和 J. M. Dolan，“利用梯形棱柱走廊和b´ézier曲线进行自动驾驶车辆的时空运动规划”，*arXiv*预印本，第arXiv–2209页，2022年。[35] J. Li、X. Xie、Q. Lin、J. He 和 J. M. Dolan，“通过在导数空间中搜索并结合凸优化进行运动规划，以扩展解空间”，发表于*2022*年*IEEE/RSJ*智能机器人与系统国际会议（*IROS*）。IEEE，2022年，第13 500–13 507页。[36] D. Dolgov、S. Thrun、M. Montemerlo 和 J. Diebel，“自动驾驶路径规划中的实用搜索技术”，安娜堡，第1001卷，第48105期，第18–80页，2008年。

Xiang Zhang 于2022年获得中国北京北京理工大学车辆工程学士学位，目前在北京理工大学机械与车辆学院智能车辆研究中心攻读硕士学位。

他的研究兴趣包括智能汽车、运动规划与控制。

Boyang Wang 于2013年获得中国北京北京理工大学车辆工程学士学位，并于2020年获得北京理工大学博士学位。2017—2019年，他曾在法国国家科学研究中心—蒙彼利埃大学LIRMM实验室交互数字人团队担任访问研究员。2020—2022年，他曾在北京大学机器感知与智能教育部重点实验室担任博士后研究员。目前，他是北京理工大学机械与车辆学院的助理教授。

他的研究兴趣包括智能汽车、驾驶员行为、运动规划与控制以及车辆动力学建模。

Yaomin Lu 于2019年获得中国北京北京理工大学车辆工程学士学位，并于2022年获得北京理工大学学位。她现任职于中国北京的百度智能驾驶事业群。

她的研究兴趣包括智能汽车、运动规划和优化方法。

Haiou Liu 于1998年和2003年分别获得北京理工大学的学士学位和博士学位。2013年9月至2014年9月，她在美国密歇根州东兰辛的密歇根州立大学机械工程系能源与汽车研究实验室担任访问学者。她现任北京理工大学机械与车辆学院教授。她的教学兴趣主要集中在本科和研究生阶段的车辆控制课程。她目前的研究兴趣包括机械式自动变速器和混合动力总成的设计与控制。

Jianwei Gong 于1992年获得中国长沙国防科技大学学士学位，并于2002年获得中国北京北京理工大学博士学位。2011年至2012年期间，他曾任麻省理工学院机器人移动研究组访问研究员。现任北京理工大学机械与车辆学院智能车辆研究中心教授兼主任。

他的研究兴趣包括智能车辆环境感知与理解、驾驶员行为以及运动规划与控制。

Huiyan Chen 于2004年获得中国北京北京理工大学博士学位。自1981年起，他一直在北京理工大学工作，并曾担任智能车辆研究中心主任。现任北京理工大学机械与车辆学院教授。

他的研究兴趣包括智能汽车、动力总成系统建模与控制、信息技术。