

doi: 10.7690/bgzdh.2026.02.017

复杂受限环境下基于图神经网络的机械臂运动规划算法

赵玉辉^{1,2}, 向学辅¹, 曾志文², 刘昆仑³, 汤立达², 王晨², 王宇倩¹

(1. 中国兵器装备集团自动化研究所有限公司系统总体部, 四川 绵阳 621000; 2. 国防科技大学智能科学学院, 长沙 410073; 3. 陆军装备部驻重庆地区军事代表局驻广元地区军事代表室, 四川 广元 628000)

摘要: 针对复杂受限环境下的机械臂运动规划对于实现普遍服务机器人至关重要的问题, 提出一种基于图神经网络 (transformer-graph neural network, T-GNN) 框架, 以解决传统基于采样规划方法的计算效率低下和现有基于学习的方法中局部依赖建模的局限性。T-GNN 集成了用于捕获全局几何依赖关系的 Transformer 模块, 并利用 GNN 进行迭代潜在图优化, 实现了高效的路径探索。实验结果表明: T-GNN 显著减少了碰撞检测次数, 提高了规划效率, 在高维场景中取得了较高的成功率, 并在不同环境复杂度下保持了路径最优性和实时性之间的良好平衡。

关键词: 图神经网络; 复杂受限环境; 碰撞检测; 运动规划; Transformer

中图分类号: TP183; TP242.6 **文献标志码:** A

Motion Planning Algorithm of Manipulator Based on Graph Neural Network in Complex Constrained Environment

Zhao Yuhui^{1,2}, Xiang Xuefu¹, Zeng Zhiwen², Liu Kunlun³, Tang Lida², Wang Chen², Wang Yuqian¹

(1. Department of System General, Automation Research Institute Co., Ltd. of China South Industries Group Corporation, Mianyang 621000, China; 2. College of Intelligence Science and Technology, National University of Defense Technology, Changsha 410073, China; 3. PLA Military Representative Office in Guangyuan District of Chongqing District Military Representative Bureau of Army Armaments Department, Guangyuan 628000, China)

Abstract: In order to solve the problem that the motion planning of manipulator in complex constrained environment is very important for the realization of universal service robot, a transformer-graph neural network (T-GNN) framework is proposed. And aim to solve that problem of low computation efficiency of the traditional sampling-based planning method and the limitation of local dependence model in the existing learning-based method. T-GNN integrates the Transformer module for capturing global geometric dependencies, and utilizes GNN for iterative latent graph optimization to achieve efficient path exploration. The experimental results show that T-GNN significantly reduces the number of collision detection, improves the planning efficiency, achieves a high success rate in high-dimensional scenes, and maintains a good balance between path optimality and real-time performance under different environmental complexities.

Keywords: graph neural network; complex constrained environment; collision check; motion planning; Transformer

0 引言

近十几年来, 机器人在工业生产和日常生活中起到了至关重要的作用。在机械臂方面, 传统的机械臂运动控制依赖固定轨迹或人工示教。不断发展的应用场景对机械臂智能化提出了更高的要求, 尤其是在冰箱等复杂受限环境中, 机械臂必须躲避障碍物并执行精确抓取任务。这种情况下的运动规划包括在开始和目标配置之间寻找一条无碰撞的路径, 能够满足路径最短、能量消耗最少等约束条件。

传统的运动规划算法主要分为图搜索和基于采样 2 类。经典的图搜索方法包括 Dijkstra 算法和基于 A* 的算法, 该类算法被广泛应用于仓库中移动机

器人的导航; 然而, 该类算法在高维状态空间具有指数级的计算复杂度, 使其不适用于多自由度机械臂。基于采样的方法主要包括快速搜索随机树 (rapidly-exploring random trees, RRT) 及其 RRT* 之类的变种算法。Wang 等^[1]提出了一种基于先验知识指导的机械臂运动规划方法, 通过改进 RRT*-Connect 算法, 利用先验知识指导双树采样过程, 实现了复杂约束环境下可行路径的快速收敛。通过构建概率树来处理规划问题, 但仍然受到维数灾难的约束, 且容易陷入局部最优解, 并生成的轨迹具有很多冗余节点, 需要进行大量的后续处理。

基于学习的方法通过数据驱动的采样优化实现了运动规划的变革。文献[2]使用条件变分自编码器

收稿日期: 2024-11-10; 修回日期: 2024-12-20

第一作者: 赵玉辉(2001—), 男, 山西人, 硕士。

(conditional variational encoder, C-VAE)来学习状态空间的分布,并据此生成高质量的路径节点。通过在潜在空间进行采样,该方法可以有效地避开障碍物区域,从而提高采样效率和路径质量。文献[3]提出了一种基于学习的拒绝采样方法,该方法通过训练一个模型来估计高概率可行域的分布;但是,正如文献[4]指出的那样,简单地将学习相关信息转换为顺序数据或潜在表示,可能无法捕获对象之间的内在联系,无法考虑关键对象之间的依赖关系。

图神经网络(graph neural network, GNN)对图中节点的顺序具有置换不变性,能够捕获对象之间的内在联系并充分考虑对象之间的依赖关系,使其成为图结构问题中学习模式的自然选择,并已被成功应用于机械臂的运动规划。文献[5]利用 GNN 来识别临界样本。文献[6]通过 GNN 从批量采样的随机几何地图中捕获几何模式,显著减少了具有挑战性的高维运动规划任务中的碰撞检查次数,提高了整体的规划效率。同时,Transformer 架构越来越多地被用于图表学习,能够有效表示图中对象的全局依赖关系。文献[7]提出了基于标准 Transformer 架构的 Graphormer 模型。该模型通过引入中心性编码、空间编码、边编码等简单有效的结构编码方式,高效地将图的结构信息融入到模型中。

基于传统 GNN 的机械臂运动规划算法难以解决长距离依赖问题,而 Transformer 在编码图空间关系时面临挑战。笔者充分利用 Transformer 和图神经网络的优势,提出了一种 T-GNN 模型架构。该方法提高了机械臂在复杂约束环境下的运动规划效率,大幅减少规划过程中碰撞检测的次数,降低了高维空间中的计算复杂度。

1 问题描述

1.1 运动规划

将机械臂的构型空间表示为 $X_{space} \subseteq R^n$, 其中 n 为构型空间的维数,一般为机械臂的自由度。构型空间 X_{space} 包括障碍物空间集合 X_{obs} 和自由空间集合 $X_{free}=X_{space}/X_{obs}$ 。

在基于采样的运动规划中,将该规划问题定义在一个随机几何图(random geometric graph, RGG)上, $G=(N, E)$ 。将采样节点设置为 $N=\{x_1, x_2, \dots, x_M\} \cup \{x_{init}, x_{goal}\}$ 且 $N \subset X_{free}$, 节点中的初始构型 x_{init} 和目标构型 x_{goal} 为属于 X_{free} 的固定节点。节点之间通过 r -disc 或者 k -NN 规则连接形成边的集合 E 。

定义一条从 x_{init} 到 x_{goal} 的路径为有限条边序列 $\pi=\{e_1, e_2, \dots, e_k\}$ 。其中, $e_i=(x_i, x_{i+1})$, $x_1=x_{init}$ 以及 $x_{i+1}=x_{goal}$ 。运动规划的核心目标是给定一个图 G 和构型对 x_{init} 和 x_{goal} , 找到一条无碰撞路径 π , 使得 π 不与障碍物相交。路径消耗代价定义为每条边的欧氏长度之和:

$$\text{cost}(\pi) = \sum_{i=1}^k |x_{i+1} - x_i|_2 \quad (1)$$

1.2 图神经网络

GNN 是整个框架的核心,用于处理由采样的构型生成的 RGG。GNN 通过采用消息传递机制在节点之间传播局部信息,从而实现无冲突边的有效优先级排序。对于图 $G=(N, E)$, GNN 通过利用其置换不变性聚合邻居信息来更新节点表示,以保证顺序独立性。具体来说, GNN 经过 k 个聚合步骤后的节点嵌入 $h_i^{(k+1)}$ 可以计算为:

$$c_i^{(k)} = \text{Aggr} \left\{ f_x \left(h_i^{(k)}, h_j^{(k)}, e_{ij} \right) \mid (x_i, x_j) \in E \right\};$$

$$h_i^{(k+1)} = \text{Combine}^{(k)} \left(h_i^{(k)}, c_i^{(k)} \right) \quad (2)$$

式中: f_x 为用于处理节点和边缘特征的多层感知器(multi-layer perceptron, MLP); Aggr 函数用于聚合邻居节点的信息。常用的聚合函数有 MEAN、Max 和 Sum, 笔者采用的是 Max 函数。Combine 函数实现了残差连接,将邻居节点的信息整合到节点表示中。通过使用相对位置和节点属性的多层感知机 h_y 对边特征 $e_{ij}=h_y(x_j-x_i, x_j, x_i)$ 进行编码,以捕获对碰撞预测至关重要的几何关系。

1.3 Transformer

在提出的 T-GNN 架构中,Transformer 主要用于全局信息的捕捉和障碍物编码。Transformer 架构主要由堆叠的 Transformer 层组成,每个层包含 2 个主要组件:自注意力模块以及位置前馈网络(feed-forward network, FFN)模块。令 $H = [h_1^T, \dots, h_n^T]^T \in R^{n \times d}$ 表示对自注意力模块的输入,其中 d 为隐藏维数, $h_i \in R^{1 \times d}$ 表示节点 i 的隐藏。输入 H 由 3 个权重矩阵 $W_Q \in R^{d \times d_k}$, $W_K \in R^{d \times d_k}$, $W_V \in R^{d \times d_v}$ 投影得到查询(Q)、键(K)和值(V)表示:

$$Q=HW_Q, K=HW_K, V=HW_V \quad (3)$$

自注意力机制表示为:

$$A = QK^T / \sqrt{d_k}, \text{Attn}(H) = \text{softmax}(A)V \quad (4)$$

式中 A 为一个矩阵,描述查询和关键字之间的相似

性。为了简单起见，在这里考虑单头自注意力，并假设 $d_k=d_v=d$ 。对多头注意力的扩展遵循标准的做法。

基于位置的前馈神经网络 FFN 对每个节点表示独立地进行非线性变换：

$$\text{FFN}(x)=\sigma(xW_1+b_1)W_2+b_2。 \quad (5)$$

式中： σ 为 ReLU 激活函数； W_1 、 W_2 为权重矩阵。笔者采用了预层归一化，在自注意力和 FFN 子层之前使用层归一化来稳定训练。

1.4 受限空间复杂度

为了量化机械臂运动规划中所处的受限环境的复杂度，采用受限空间复杂度 (constrained space complexity, CSC)^[1]，综合影响路径规划的多个因素。该度量通过提供一个系统的量化模型来解决定性描述的模糊性。

将机械臂的工作空间定义为半径为 R 的球面区域，其基坐标 $B(0, 0, 0)$ 偏离球心，计算位于 B 点下方的地面区域 W_{ground} 的体积为：

$$W_{\text{ground}}=\frac{\pi}{3}h^2(3R-h)。 \quad (6)$$

式中 h 为球心到地平面的垂直距离。排除地面的有效工作空间体积 W_{eff} 为：

$$W_{\text{eff}}=\frac{4}{3}\pi R^3-W_{\text{ground}}。 \quad (7)$$

障碍物引起的复杂度首先通过障碍物总体积与有效工作空间的比值进行量化：

$$\xi_1=\frac{\sum_{i=1}^m O_{\text{vol}}^i}{W_{\text{eff}}}。 \quad (8)$$

式中： O_{vol}^i 为第 i 个障碍物的体积； m 为障碍物的个数。为了解决空间分布效应，对初始和目标构型的约束影响被建模为：

$$\xi_2=V_{\text{init}}/C_{\text{init}}+V_{\text{goal}}/C_{\text{goal}}。 \quad (9)$$

式中： V_{init} 和 V_{goal} 分别为 x_{init} 和 x_{goal} 周围的约束体积； C_{init} 和 C_{goal} 分别为包围这些构型的凹形障碍物体积，如图 1 所示。基于构形间的直接路径定义了宏观阻塞因子 ξ_3 ：

$$\xi_3=\begin{cases} 0, & x_{\text{init}}x_{\text{goal}} \cap O_{\text{vol}}=\emptyset \\ 0.5, & x_{\text{init}}x_{\text{goal}} \cap O_{\text{vol}} \neq \emptyset \end{cases}。 \quad (10)$$

综合 ξ_1 、 ξ_2 、 ξ_3 ，将模型的复杂性表示为：

$$\xi=\max(\xi_1+\xi_2+\xi_3, 1)\times 100\%。 \quad (11)$$

该指标综合障碍物密度、构型约束和直达路径

干扰，定量反映规划难度。

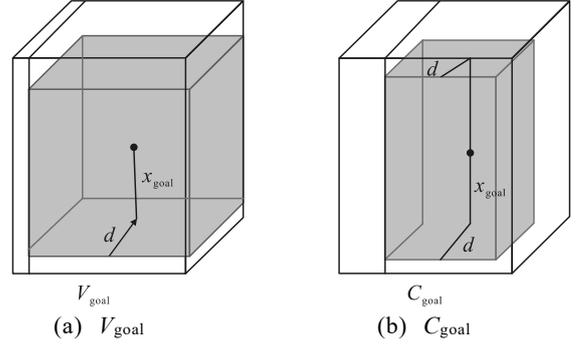


图 1 目标 V_{goal} 和 C_{goal}

2 方法

2.1 总体方法

一般而言，运动规划中的批量采样方法通常遵循以下步骤：1) 从自由空间中采样一组状态样本，通过 k -NN 等算法构建随机几何图 RGG；2) 采用启发式搜索策略，在图内构建从起始点逐步扩展到目标点的搜索树，由于树形结构只允许零碰撞条边，每次连接操作都需要进行碰撞检测；3) 当识别出一条可行路径后，系统将其记录为候选解，并不断优化路径以降低成本；4) 在完成当前批次内所有无碰撞边的整合后，引入新批次样本继续扩展搜索树；5) 该算法迭代采样-扩展过程，直到达到预先设定的计算资源限制，最终输出当前最优路径或返回失效状态。

笔者提出的 T-GNN 框架旨在优化规划过程中的路径探索模块，如图 2 所示。该框架将 Transformer 模块与 GNN 相结合，利用全局几何依赖捕获和迭代隐层图优化机制实现高效的路径探索。通过目标导向的注意力机制选择无碰撞的边缘，并采用基于多头注意力的障碍物感知编码，加强环境障碍物的特征表达和感知能力，从而减少碰撞检测的边缘数量，降低计算复杂度。对 Transformer 编码过的节点、边以及障碍物进行建模，利用 GNN 来进行不断迭代嵌入，充分收集邻居信息，捕获建模后的图的对象之间的内在联系。输出策略矩阵，得到最佳路径。

2.2 用于路径规划的 T-GNN 架构

T-GNN 架构以批量采样生成的采样随机几何图形 $G=(N, E)$ 作为输入。每个顶点 x 包含一个 n 维构型空间坐标和一个 3 维 one-hot 标签，分别代表自由空间中的顶点、有碰撞的顶点和特殊的目标顶点。

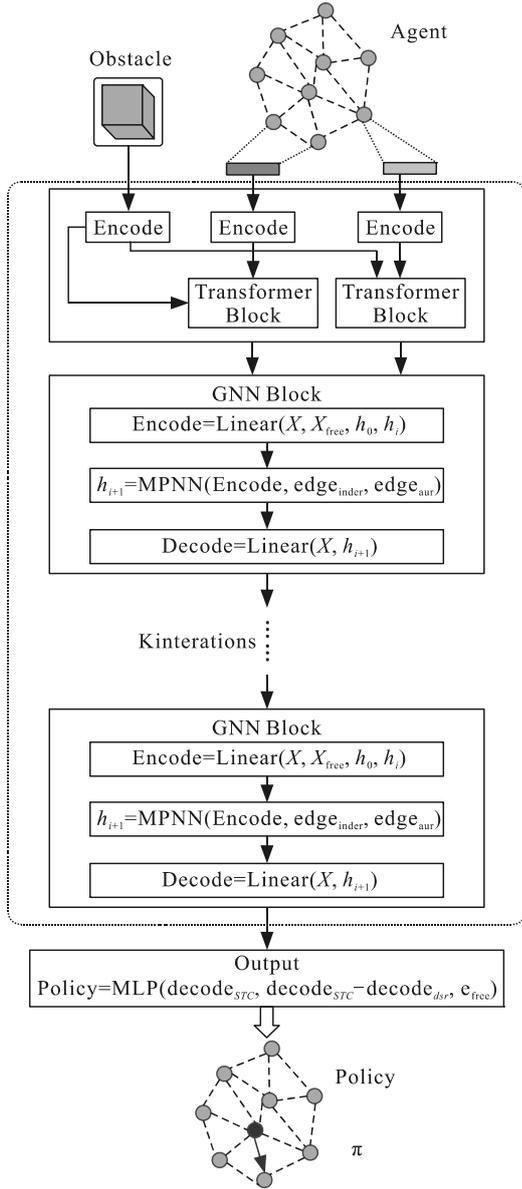


图 2 基于 T-GNN 算法的机械臂运动规划框架

将顶点和边嵌入到一个隐空间中，其中 $v \in R^{|E| \times d_h}$ ， $e \in R^{|E| \times d_h}$ ， d_h 是嵌入的大小。取顶点和边的嵌入 v 和 e ，GNN 通过 2 个多层感知机 f_x 和 f_y 来聚合每个顶点来自其邻居的局部信息，如式(12)所示。

$$v_i = g(v_i, \max\{f_x(v_j - v_i, v_j, v_i, e_l)\});$$

$$e_l = \max\{e_l, f_y(v_j - v_i, v_j, v_i)\}, (v_i, v_j) \in E. \quad (12)$$

式中： $v_i = \text{PosEnc}(h_x(x_i, x_g, (x_i - x_g)^2, x_i - x_g))$ 为生成的 RGG 图中顶点嵌入的形式， h_x 为一个带有批量归一化的 2 层 MLP。笔者利用 Transformer 的位置编码 PosEnc 将空间位置信息融入到节点中，缓解了在高维空间中表示几何关系的挑战。将 L_2 距离和目标点到顶点嵌入的距离之差加入到顶点嵌入中。这

为 GNN 更好地识别价值更高的节点提供了启发。对于边 e_l 的计算方法是 $e_l = h_y(x_j - x_i, x_j, x_i)$ ，其中 h_y 也是 2 层 MLP。经过多次迭代后，策略网络输出每条边的优先级 $\eta_l = f_\eta(y)$ 。

此外，为处理可变障碍物 $O = \{o_1, \dots, o_m\}$ ，模型采用多头注意力将障碍物的信息融入节点的嵌入，如图 3 所示。

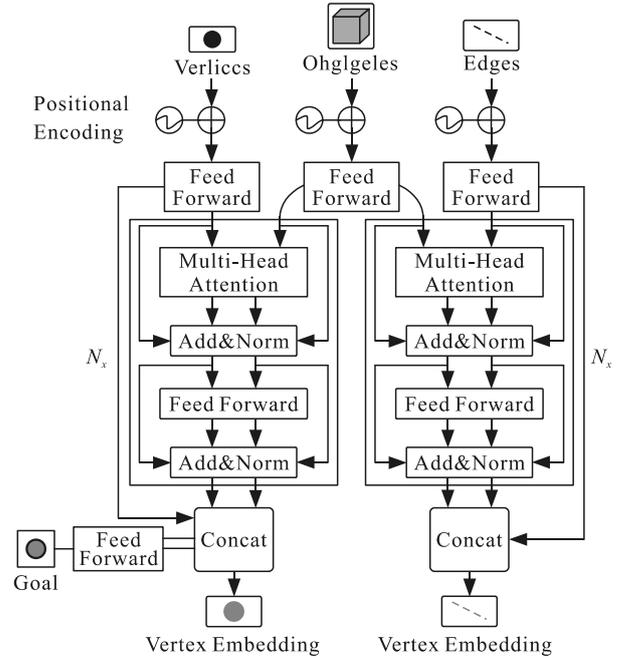


图 3 基于 Transformer 的运动规划编码模块架构

模型可以通过利用全局的障碍-节点交互来优先考虑距离障碍物较远的边：

$$h_i^{\text{obs}} = \text{MultiHead}(h_i, O). \quad (13)$$

式中： h_i 为节点的嵌入； O 为多头注意力机制的键和值。注意力机制保证了排列不变性，使其对不同的障碍物顺序具有鲁棒性。

在 GNN 消息传递过程中，对融合障碍物的节点嵌入集合 h_i^{obs} 进行聚合：

$$c_i^{(k)} = \max\{f_x(h_i^{\text{obs}}, h_j^{\text{obs}}, e_{ij}) | (v_i, v_j) \in E\}. \quad (14)$$

这样可以使模型能够避开碰撞易发区域。比如在 U 型障碍物场景中，多头注意力可以捕获边界特征，优先考虑绕过障碍物的边缘，减少不必要的碰撞检查。

3 实验分析

3.1 实验设置

本实验采用 Coppeliassim 4.3.0 作为实验仿真平台，所有的训练和测试实验均在一台搭载 Intel i7-14700KF CPU @ 3.40 GHz 的计算机上进行，该

计算机还配备单张 NVIDIA RTX 4070 GPU。笔者采用 PyTorch 框架进行模型构建,在模型训练过程中,笔者选择 Adam 优化器进行训练。

3.2 结果分析

3.2.1 训练结果分析

将笔者提出的 T-GNN 方法与 5 种规划算法进行了比较,包括基于采样的规划 RRT*、基于批量采样的方法 BIT*、基于懒惰运动规划的方法 LazySP、基于学习的方法 NEXT 以及基于 GNN 的方法。在实验中,随机生成了 2 000 个用于模型训练的数据集和 500 个用于模型测试的数据集。每个问题包含一组随机分布的障碍物和一对可行的起始点 x_{init} 和目标点 x_{goal} 。所有实验都用 4 种不同的随机种子重复进行,最终的平均结果如表 1 所示。

表 1 T-GNN 算法与其他方法的比较

规划方法	成功率/%	碰撞检测 次数	路径 代价/m	规划 时间/s
RRT*	46.4	3 065.94	5.39	0.40
BIT*	100.0	2 468.77	10.04	0.28
LazySP	99.6	1 706.85	11.00	0.27
NEXT	41.6	6 349.14	4.76	5.43
GNN	100.0	2 192.84	11.58	0.17
GNN+Smoother	100.0	6 078.31	8.78	0.33
T-GNN(Ours)	100.0	1 266.29	11.26	0.16
T-GNN+Smoother(Ours)	100.0	4 781.87	8.58	0.31

实验结果表明,笔者提出的方法在综合考虑的情况下优于 RRT*, BIT*, LazySP, NEXT 和 GNN。

在成功率方面,本文中方法达到了 100%,与 BIT*和 GNN 方法处于领先水平,而 RRT*和 NEXT 的成功率仅为 46.4%和 41.6%,明显低于 T-GNN 算法。

碰撞检查次数是衡量算法效率的关键指标。笔者提出的 T-GNN 算法平均只需要 1 266.29 次,比 BIT*减少了 59.4%,比 NEXT 算法减少了 78.3%,很大程度上少了不必要的计算,显示了其高效避障和路径规划的能力。与原始的 GNN 方法相比较,平均进行 2 192.84 次的碰撞检查,本文中方法减少了 926.55 次,减少比例为 40%,表明笔者提出的 T-GNN 算法能够更准确地规划路径,有效减少与障碍物的交互计算,在效率上明显优于 GNN。

在规划时间维度上,T-GNN 用 0.16 s 完成路径规划,优于大多数其他算法。虽然原始 T-GNN 的路径成本为 11.26,略高于 NEXT,但结合路径平滑策略后,成本可降低至 8.58,实现了效率与轨迹质量的平衡。与原始的 GNN 方法相比,T-GNN 不仅减少了 40%的碰撞检测次数,而且显著缩短了路径

规划时间,证实了其多方面的优越性。

3.2.2 仿真结果分析

在 Coppeliassim 平台上设置了 2 个场景进行对比实验,以评估 RRT*、RRT*-Connect、GNN 和 T-GNN 算法,如图 4—5 所示。

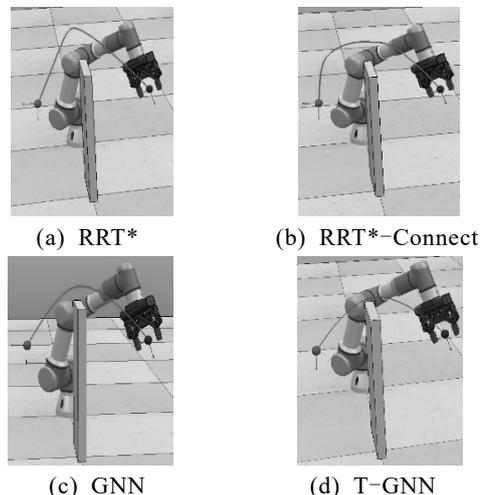


图 4 机械臂在场景 1 中不同算法的规划过程

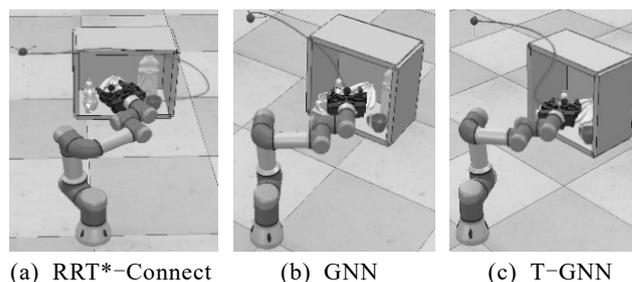


图 5 机械臂在场景 2 中不同算法的规划过程

在图 4—5 中,左侧小球代表起点,右侧小球代表终点,连接 2 个小球的线段代表各个算法生成的规划路径。在 3.3 节中笔者设置了更多复杂的场景来进行成功率验证。场景 1 中,在起始点和目标点之间放置简单障碍物,约束空间复杂度(CSC)为 50.31%;场景 2 的特点是在目标周围也设置障碍物,CSC 高达 67.73%,RRT*算法在该场景下规划失败。在基于搭载 Robotiq2F-85 夹爪的 UR3 机械臂进行自主避障实验,所有实验均用 10 种不同的随机种子重复进行,最终的实验平均结果如表 2 所示。

表 2 2 种场景下 4 种方法的比较

场景	规划方法	碰撞检测次数	路径代价/m	规划时间/s
1	RRT*	1 072.32	2.75	0.72
	RRT*-Connect	1 410.64	5.13	0.61
	GNN	1 062.16	2.95	1.20
	T-GNN(Ours)	324.56	2.80	0.40
2	RRT*	10 000.00	N/A	300
	RRT*-Connect	3 394.27	9.50	3.47
	GNN	1 777.68	5.47	3.01
	T-GNN(Ours)	1 169.60	11.16	2.00

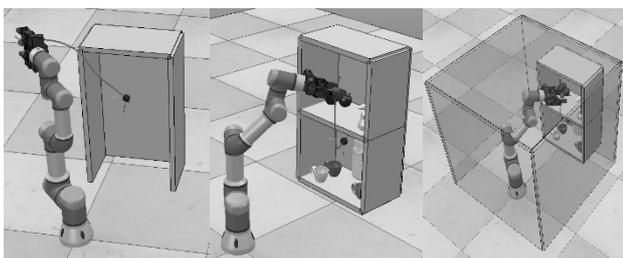
在场景 1 中，本文中方法通过障碍物感知编码构建的全局交互机制实现了最低的碰撞检测次数和规划时间，路径代价为 2.80，接近 RRT* 的 2.75。该机制通过多头注意力整合障碍物信息，使模型能够动态捕捉环境的几何依赖关系，避免无效探索。在复杂场景 2 中，本文中方法在 2.00 s 的规划时间和 1 169.60 次的碰撞检查中完成了路径搜索，成功率为 100%，而 RRT* 在 10 000 次检查后失败。虽然 GNN 路径较短，但本文中方法通过 Transformer 增强的长距离依赖建模，优先探索具有目标导向注意力的无碰撞边，与 GNN 相比减少了 34.2% 的碰撞检查次数，突出了复杂受限环境下路径发现的高效性。

本文中方法通过障碍物编码和全局特征交互的协同设计，在复杂受限环境下实现了场景 1 比 RRT* 减少 70.7% 的碰撞检查，场景 2 比 GNN 缩短 33.5% 的规划时间和路径质量的综合优化，并且在传统算法失效情况下的高维约束空间中依然保持 100% 的成功率。展现出很好的鲁棒性和环境适应能力。

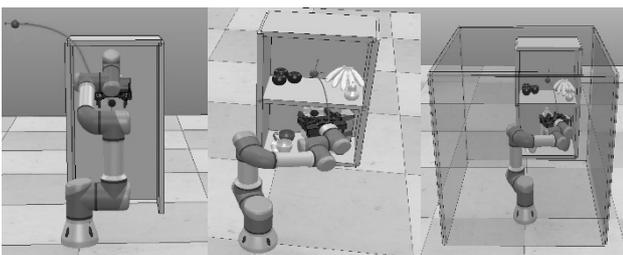
3.3 规划成功率验证

为了验证笔者提出的算法在高 CSC 条件下可以保证较高的成功率，构建了 CSC 值不断增加的场景 3、场景 4 和场景 5，CSC 值分别为 83.54%，94.31%，95.56%，以评估不同 CSC 等级下的规划成功率和规划时间。笔者将规划成功率定义为可行路径的实验次数的比值为成功找到的实验总数。

为了全面评估所提出方法的有效性和鲁棒性，在每种测试场景下均进行了 10 次自主避障实验，如图 5 所示。



(a) 机械臂起始构型



(b) 机械臂目标构型

图 5 其他场景的运动规划结果

图 5 给出了这 3 种场景下的机械臂运动规划结果，验证了机械臂能够无碰撞地到达目标点。实验的平均数据统计如图 6 所示。

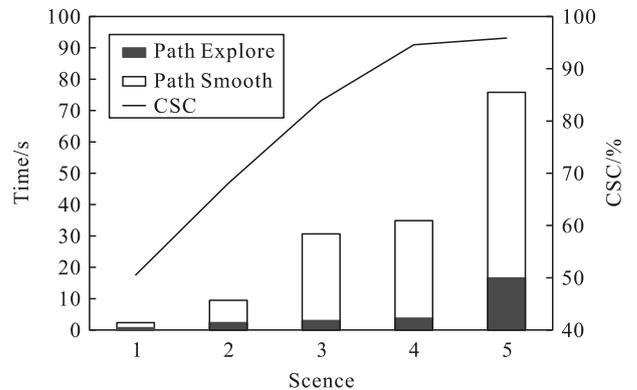


图 6 不同场景的运动规划时间及对应的 CSC

值得注意的是，即使在 CSC 为 95.56% 的情况下，尽管路径规划所需的时间增加到了 75.15 s，但该算法依然能够保持 100% 的成功率。这一结果充分表明，该算法在保证任务成功率的同时，也具备良好的计算效率和稳定性，体现了在成功率与计算成本之间的良好平衡。

4 结论

笔者提出的 T-GNN 算法在复杂受限约束环境下的机械臂运动规划中表现出明显的优势。通过将 Transformer 和图神经网络相结合，该方法克服了传统基于采样的方法的计算效率低下和现有基于学习的方法在局部依赖建模方面的局限性，实现了高成功率、高效的路径探索，并在路径最优性和实时性之间取得了良好的平衡。未来的工作将侧重于集成动态避障和实时传感器融合，将该框架扩展到非结构化环境中并运用到实物机械臂。

参考文献：

- [1] WANG X, ZHU D, GUO C, et al. Manipulator Motion Planning Based on Prior Knowledge Guidance in Complex Constrained Environment[C]//2023 42nd Chinese Control Conference (CCC). Tianjin, China: IEEE, 2023: 4872-4879.
- [2] ICHTER B, HARRISON J, PAVONE M. Learning Sampling Distributions for Robot Motion Planning[C]//2018 IEEE International Conference on Robotics and Automation (ICRA). Brisbane, QLD: IEEE, 2018.
- [3] ZHANG C, HUH J, LEE D D. Learning Implicit Sampling Distributions for Motion Planning[C]//2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Madrid: IEEE, 2018: 3654-3661.