

# 基于 DLGS-RRT-Connect 算法的 狭窄复杂空间路径规划<sup>\*</sup>

薛佳慧 陶贤露 潘树国 王萍 高旺 张俊豪

(东南大学仪器科学与工程学院 综合时空网络与装备技术全国重点实验室 南京 210096)

**摘要:**路径规划是无人车辆实现自主导航的关键技术,决定着无人车辆能否安全高效地抵达目标点。但是,常见的路径规划算法在狭窄通道环境中存在收敛速度慢、规划耗时长以及路径质量差的问题。为此,提出一种基于双层引导采样的 RRT-Connect(DLGS-RRT-Connect)算法。首先,在狭窄通道中预先构建引导路径,采用搜索式连接策略引导随机树在狭窄通道中沿引导路径拓展,从而减少无效采样,提升算法在狭窄通道内的探索效率。其次,算法引入目标偏置策略,降低采样过程中的随机性,为随机树的生长提供方向性引导,从而进一步提升路径规划的效率。最后,仿真结果表明,相较于常见的 Goal\_bias RRT、Informed-RRT<sup>\*</sup> 和 RRT-Connect 算法,DLGS-RRT-Connect 算法在狭窄通道环境中的规划成功率分别提高了 35%,60%,26%,平均规划时长分别降低了 70.62%,97.65%,63.92%,路径平均长度也分别减少了 14.53%,16.70%,18.84%,可以有效改善狭窄环境规划路径的平滑性和安全性。

**关键词:**DLGS-RRT-Connect; 快速搜索随机树(RRT); 引导路径; 目标偏置采样; 路径优化

**中图分类号:** TP242; TN96      **文献标识码:** A      **国家标准学科分类代码:** 510.80

## Narrow and complex spatial path planning based on DLGS-RRT-Connect algorithm

Xue Jiahui Tao Xianlu Pan Shuguo Wang Ping Gao Wang Zhang Junhao

(School of Instrument Science and Engineering, State Key Laboratory of comprehensive PNT Network and  
Equipment Technology, Southeast University, Nanjing 210096, China)

**Abstract:** Path planning is a key technology for unmanned vehicles to realize autonomous navigation. Whether a safe and smooth travelable path can be quickly planned in a narrow channel determines the efficiency of unmanned vehicles in performing tasks in narrow and complex environments. However, common path planning algorithms usually have the problems of slow convergence speed, long planning time and poor path quality in the narrow channel environment. For this reason, this paper proposes a RRT-Connect algorithm Based on dual-layer guided sampling (DLGS-RRT-Connect) algorithm. First, the guided path is pre-constructed in the narrow channel, and the searching connection strategy is used to guide the random tree to expand along the guided path in the narrow channel, so as to reduce the invalid sampling and improve the exploration efficiency of the algorithm in the narrow channel. Secondly, the algorithm introduces a target bias strategy to reduce the randomness in the sampling process and provide directional guidance for the growth of the random tree, thus further improving the efficiency of path planning. Finally, the simulation results show that compared with the common Goal\_bias RRT, Informed-RRT<sup>\*</sup>, and RRT-Connect algorithms, the DLGS-RRT-Connect algorithm proposed in this paper improves the planning success rate in narrow channel environments by 35%, 60%, and 26%, respectively, and reduces the average planning time by 70.62%, 70.62%, 70.65%, and 97.65%, and 63.92%, and the average path length is also reduced by 14.53%, 16.70%, and 18.84%, respectively, which can effectively improve the smoothness and safety of planning paths in narrow environments.

**Keywords:** DLGS-RRT-Connect; RRT; guided paths; target bias sampling; path optimization

## 0 引言

随着智能无人技术的不断进步,地下管网、矿井隧道、灾后救援及自动化仓储等狭窄复杂环境中智能无人车辆的应用正日益受到关注,并逐渐成为提升作业效率与安全性的关键途径<sup>[1-3]</sup>。在智能无人车辆的自主导航中,路径规划至关重要,它决定了车辆如何在复杂环境中高效、安全地到达目标位置<sup>[4-5]</sup>。

路径规划旨在配置空间中找到一条从给定起点到目标点的无碰撞路径,并尽可能缩短路径长度、提高平滑性、减少计算耗时,同时满足车辆的运动约束<sup>[6-8]</sup>。目前,常用的全局路径规划算法有 A\* 算法、快速拓展随机树算法(rapidly-exploring random trees, RRT)、遗传算法、蚁群算法等<sup>[9]</sup>。其中,RRT 算法因其建模简单、具有概率完备性且采样搜索高效,近年来成为解决高维规划问题的主流方法<sup>[10]</sup>。然而,由于其随机采样特性,在狭窄复杂空间中难以获得有效采样点导致搜索效率降低,规划路径蜿蜒曲折<sup>[12-13]</sup>。

针对上述问题,国内外学者提出了许多改进措施<sup>[14]</sup>。Kuffner 等<sup>[15]</sup>提出了 RRT-Connect 算法,在规划过程中采用双向树生长与贪婪连接启发式,有效实现了搜索树的双向快速扩展。为了提高采样效率,减小采样的随机性,Lavalle 等<sup>[16]</sup>提出了 Goal\_bias RRT,将目标点以一定概率设置为采样点,以此引导随机树的生长。Karman 等<sup>[17]</sup>提出了 RRT\* 算法,在传统 RRT 算法的基础上引入了父节点重选择机制,在采样的过程中不断优化采样点与随机树的连接距离,在保持在线规划效率的同时实现规划路径的渐进最优,提高了路径质量。为了进一步提高采样效率,Gammell 等<sup>[18]</sup>提出的 Informed-RRT\* 算法将采样区域限制在一个椭球体中,而不对整个空间进行随机采样。随着采样次数的迭代逐步缩小采样区域,同时不断更新采样点与随机树的连接关系,从而更加高效地实现规划路径的渐进最优。此外,Liao 等<sup>[19]</sup>提出一种 F-RRT\* 算法,基于三角形不等式创建新节点,并结合父节点重选择策略,从而优化路径成本,显著提高了初始解的质量和收敛速度。Li 等<sup>[20]</sup>提出的 PQ-RRT\* (potential quick-RRT\*) 算法通过引入基于势场的采样策略引导采样点向目标点靠近,并在父节点重选择时将搜索范围扩大至祖先节点,从而优化了初始解的质量和收敛速度,同时保持了渐进最优性。

经过众多专家学者的深入研究,改进的 RRT 算法在采样效率、收敛速度以及路径质量等方面已经获得显著提升。然而,在狭窄复杂的环境中,RRT 算法及其改进算法由于难以获得有效采样点,仍然难以高效规划出平滑路径。针对 RRT 及其改进算法在狭窄复杂空间中存在

的规划效率低、收敛速度慢及路径质量差等问题,本文在 RRT-Connect 算法的基础上提出了一种基于双层引导采样的 RRT-Connect (RRT-Connect algorithm based on dual-layer guided sampling, DLGS-RRT-Connect) 算法。首先,使用快速桥接法提取狭窄通道特征点,对特征点进行聚类分析并通过构建最小生成树的方式构建狭窄通道引导路径。然后,通过狭窄通道引导路径以及目标偏置策略对随机树的生长方向进行双层引导,加速随机树在狭窄通道中的拓展,提高初始路径的规划效率。随后,通过贪婪剪枝、转角优化以及四阶贝塞尔曲线平滑的方式对路径进行了优化处理。

## 1 RRT 算法及其改进算法

### 1.1 传统 RRT 算法

RRT 是一种基于采样的路径规划算法,其核心思想是通过在自由空间随机采样,并在可行区域拓展随机树,不断探索环境,最终找到从起点到目标点的可行路径。

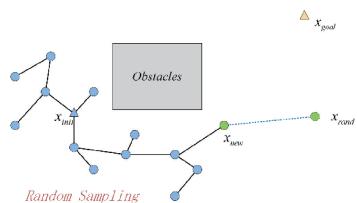
如图 1 所示,传统 RRT 算法的实现步骤如下:

- 1) 在自由空间中进行随机采样得到点  $x_{rand}$ ;
- 2) 若是  $x_{rand}$  不在障碍物中,则计算随机树上离  $x_{rand}$  最近的点  $x_{near}$ ;
- 3) 若是  $x_{near}$  与  $x_{rand}$  的距离小于拓展步长  $step$ ,则将  $x_{rand}$  作为新节点  $x_{new}$ ,否则从  $x_{near}$  向着  $x_{rand}$  的方向拓展一个步长  $step$  得到新节点  $x_{new}$ ;
- 4) 判断随机树与新节点  $x_{new}$  的连接是否会与障碍物碰撞,若是不碰撞则将  $x_{new}$  加入随机树中,反之则返回步骤(1)重新进行采样;
- 5) 计算随机树的新节点  $x_{new}$  与目标点  $x_{goal}$  的距离,若是两点距离小于  $d$  则代表随机树可以与目标点  $x_{goal}$  相接,路径搜索完成,反之则重复上述步骤,通过不断采样迭代拓展随机树的搜索范围。

### 1.2 RRT-Connect 算法

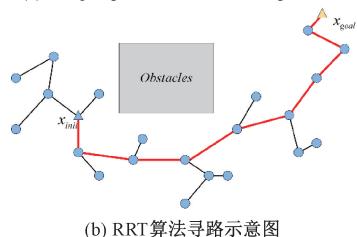
RRT-Connect 算法在 RRT 的随机采样基础上,从起点和目标点分别生成两棵随机树,并引入启发式拓展策略,通过动态切换拓展方式,使两棵树相连,从而生成起点至目标点的全局路径。

如图 2 (a) 所示,算法以规划起点  $x_{init}$  和目标点  $x_{goal}$  为根节点同时进行两棵随机树的拓展。在一个生长周期中,一颗随机树在可行驶区域进行随机采样得到  $x_{rand}$ ,选择该随机树上距离  $x_{rand}$  最近的点作为生长点,从该点向  $x_{rand}$  的方向以一定步长  $step$  进行拓展得到一个新的节点  $x_{new1}$ ,若是生长点到  $x_{new1}$  的拓展过程中不与障碍物碰撞,则完成一次随机采样拓展。此时,另一棵随机树选择树上距离  $x_{new1}$  最近的点作为生长点进行启发式拓展,从该



(a) RRT算法采样示意图

(a) Sampling schematic of RRT algorithm



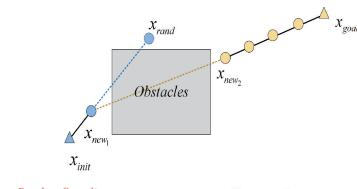
(b) RRT算法寻路示意图

(b) Schematic diagram of RRT algorithm pathfinding

图 1 RRT 算法原理

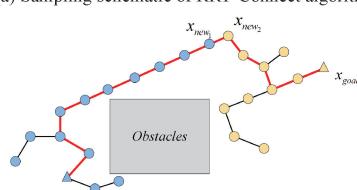
Fig. 1 Principle of RRT algorithm

点向  $x_{new1}$  方向不断以步长  $step$  进行拓展得到新节点  $x_{new2}$ , 直到随机树在该方向上与障碍物相撞或者  $x_{new1}$  和  $x_{new2}$  相遇。考虑到两棵树的平衡性, 在不同拓展周期动态切换两树的拓展策略, 促使两棵随机树更快相接。如图 2(b) 所示, 当  $x_{new1}$  和  $x_{new2}$  相遇时两棵随机树连接成为航路树, 在航路树上进行回溯即可得到从规划起点到目标点的全局路径。



(a) RRT-Connect 算法采样示意图

(a) Sampling schematic of RRT-Connect algorithm



(b) RRT-Connect 算法寻路示意图

(b) Schematic diagram of RRT-Connect algorithm pathfinding

图 2 RRT-Connect 算法原理

Fig. 2 Principle of RRT-Connect algorithm

相较于 RRT 算法, RRT-Connect 通过双向拓展以及贪婪拓展策略极大地提高了规划效率。但在具有狭窄通道的环境中, RRT-Connect 算法仍然存在难以取得有效采样点的缺陷, 且其规划路径的质量仍无法满足无人车辆正常行驶的需求。

## 2 DLGS-RRT-Connect 算法

传统 RRT 及 RRT-Connect 算法在狭窄通道中难以获得足够的有效采样点, 限制了随机树在复杂环境中的拓展能力。为提高采样效率, 本文提出了一种基于狭窄通道引导路径及目标偏置引导的双层引导采样方法, 极大程度上优化了算法在狭窄通道内的搜索过程。最后, 本文通过对初始路径进行优化最终生成了安全且平滑的可行路径, 算法框架如图 3 所示。

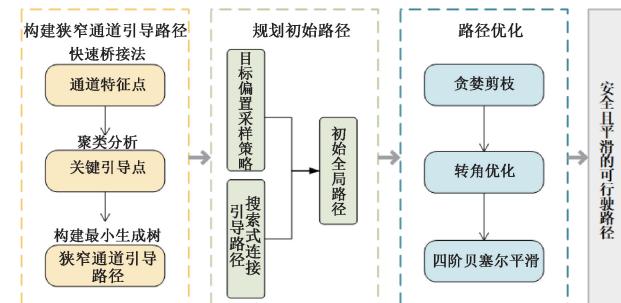


图 3 DLGS-RRT-Connect 算法框架图

Fig. 3 DLGS-RRT-Connect algorithm framework diagram

### 2.1 构建狭窄通道引导路径

引导路径为随机树在狭窄通道中的拓展提供引导作用, 使得随机树能够快速拓展通过狭窄通道。本文首先通过快速桥接法提取狭窄通道的特征点, 然后利用 K-means 算法对特征点进行聚类分析, 所提取到的关键点能够准确识别并概括狭窄通道中的重要结构特征。最后, 通过两次构建最小生成树的方式构建狭窄通道引导路径。

#### 1) 通道特征点提取

一般的桥接法通过随机采样找到位于障碍物中的点, 在该点附近不断进行高斯采样直到在通道中搭建有效桥梁。这种提取通道特征点的方式效率较低, 且易于在通道拐弯处错误搭建桥梁, 从而获取错误的通道特征点, 误导后续的路径规划过程。为高效提取狭窄通道特征点, 本文通过贪婪连接的方式构建狭窄通道桥梁网络, 并使用正交检测法对所构建桥梁进行筛选, 从而避免了在凹陷拐角处获得错误特征点。通道特征点的提取方式如下。

(1) 桥梁构建。如图 4 所示, 在地图中随机撒点, 将处于障碍物中的采样点放入到集合  $Need\_node$  中。将  $Need\_node$  中的采样点两两组合, 若是两点  $x_1, x_2$  间距离在一定范围内, 且两点连线的中点  $x_{mid}$  不在障碍物中, 则判定  $x_1, x_2$  能够组成桥梁。

(2) 桥梁筛选。对桥梁进行正交测试, 如图 5 所示,

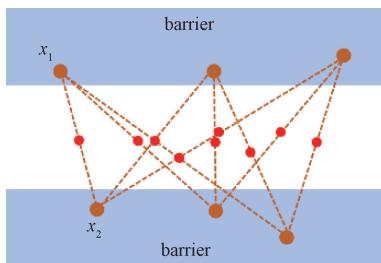
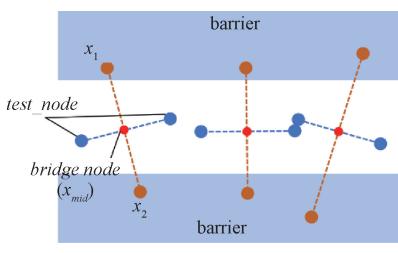


图 4 构建狭窄通道桥梁网络

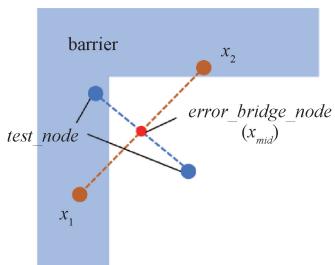
Fig. 4 Building a network of narrow passage bridges

取  $x_1, x_2$  连线的垂直平分线上与  $x_1, x_2$  连线中点一定距离的两点  $test\_node_1, test\_node_2$  若是它们都不在障碍物内, 则判定  $x_1, x_2$  组成的桥梁不位于拐角处, 桥梁有效。

(3) 通道特征点提取。筛选得出所有有效桥梁, 取桥梁中点得到狭窄通道特征点集合  $bridge\_node$ 。



(a) 获取正确通道特征点  
(a) Getting the right channel feature point



(b) 筛除错误通道特征点  
(b) Screening out erroneous channel feature points

图 5 正交检测

Fig. 5 Orthogonal detection

如图 6 所示, 快速桥接法能够在进行相同数量采样的情况下获得更多桥梁, 提高桥梁生成效率, 同时能够避免拐角处生成错误特征点, 以防影响后续狭窄通道引导路径的构建。

使用快速桥接法获得的通道特征点较为集中地分布在通道中线附近, 能够较高程度地表征狭窄通道的可通行区域。但是, 过多冗余的特征点会给后续规划过程带来不必要的计算量, 降低规划效率。

由图 6 可知, 通道特征点既有密集分布区域, 也存在孤立特征点。采用聚类分析方法能够很好地提取出最能反映狭窄空间结构特性的引导点。为了减少重合度较高

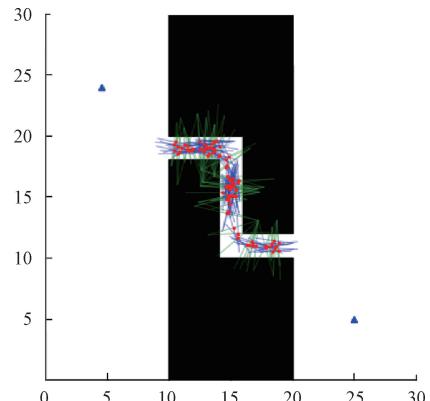


图 6 快速桥接法获取通道特征点

Fig. 6 Fast bridging method to obtain channel feature points

的冗余特征点, 本文利用 K-means 算法对获取的通道特征点进行聚类分析。首先在特征点中随机选择一定数量的点作为类簇中心, 通过计算各特征点与类簇中心的距离对特征点进行聚类划分, 距离越近则相似度越高。随后通过计算簇内所有特征点位置的均值以更新簇中心的位置, 更新位置后的簇中心更加贴近簇内特征点的真实分布中心。最后重复执行“特征点重分配”及“簇中心重计算”两个步骤直到簇内距离的平方和 (SSE) 不再显著下降, 簇内距离的平方和为:

$$SSE = \sum_{i=1}^K \sum_{x \in C_i} \|x - \mu_i\|^2 \quad (1)$$

式中:  $K$  是簇的数量;  $C_i$  是第  $i$  个簇;  $x$  是簇  $C_i$  中的数据点;  $\mu_i$  是簇  $C_i$  的中心。

通过不断调整簇中心的位置和重新分配特征点, K-means 算法能够找到一个局部最优的解, 使得狭窄通道特征点与最终得到的簇中心的簇内距离平方和最小。由此得到的簇中心既可以很好地概括狭窄通道的空间特性, 适合作为狭窄通道的引导点。如图 7 所示, 通过 K-means 算法提取的引导点能够有效代表狭窄通道可行驶区域。

## 2) 构建引导路径

通过聚类分析得到的狭窄通道引导点虽能够表征狭窄通道的可行驶区域的空间结构特性, 但不具备对可行驶区域连通方向的描述。构建狭窄通道引导路径能够最大程度地概括可通行区域的特征, 对后续随机树在狭窄通道中的拓展提供方向性的引导。

本文利用狭窄通道引导点构建最小生成树来生成引导路径, 引导路径的构建过程如图 8 所示。图论中将联通所有点的无向图称为生成树, 构建狭窄通道引导点的最小生成树能够找到通道中最短的可行驶路径<sup>[21]</sup>。

Prim 算法是一种用于寻找无向图的最小生成树的

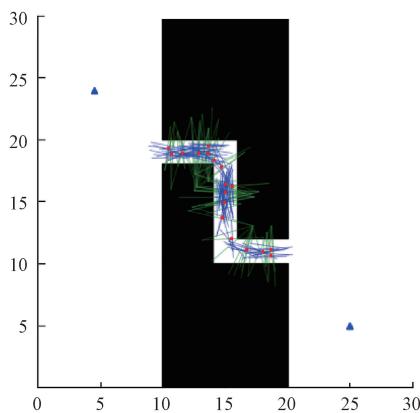


图 7 聚类后通道特征点分布情况

Fig. 7 Distribution of channel feature points after clustering



图 8 引导路径构建流程

Fig. 8 Guided path building process

算法。算法从任意点开始,寻找连接树中节点的权重值最小的边,若该边的另一端不在树中则将该边加入生成树,反之则从余下的边中寻找最小的边进行添加,直到所有顶点都在生成树中。如图 9 所示,首先以两引导点间距离为权重,构建狭窄通道引导点的无向图,同时对无向图中的边进行碰撞检测,仅保留引导点间的安全连接关系。使用 Prim 算法寻找无向图的最小生成树,并在构建过程中加入碰撞判断,可以提取出一条较短的狭窄通道引导路径。

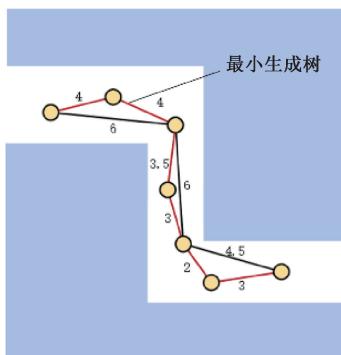


图 9 寻找无向图的最小生成树

Fig. 9 Finding the minimum spanning tree of an undirected graph

由于在通道交叉路口形成通道引导点的概率较低,受碰撞条件约束难以通过构建一次最小生成树去连接各通道分支。当通道狭长,特征点距离过远时用于引导路径与拓展树连接的识别点数量不足,加大了拓张树与引

导路径的连接难度。于是,本文对生成树较长的树枝进行插值,增添通道引导点,并二次构建最小生成树,尽可能使用一条引导路径概括相通的狭长通道,通道引导路径构建结果如图 10 所示。通过上述方法提前构建通道引导路径能够避免在狭窄通道内进行路径规划,提高含有通道场景的规划成功率。

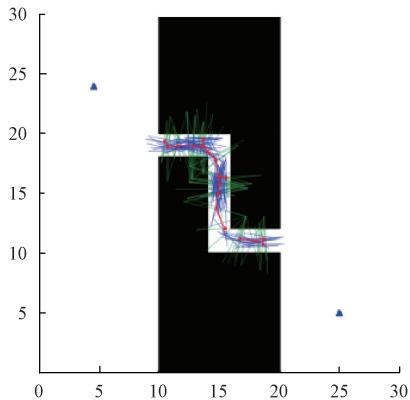


图 10 构建狭窄通道引导路径

Fig. 10 Constructing narrow passage guided paths

## 2.2 双层引导采样的 RRT-Connect 算法

### 1) 目标引导采样策略

随机树通过随机采样的方式进行拓展虽能快速探索环境,但因采样缺乏方向性,导致在非目标区域进行了大量的无效采样,降低了路径搜索效率。针对此问题,本文考虑在采样过程中加入目标引导策略,以一定概率  $P$  将非采样随机树的根节点作为采样点,以  $(1-P)$  的概率通过随机采样得到采样点,为拓展树的生长方向提供指导。

$$x_{rand} = \begin{cases} x_{goal}, & P \leq P_{goal} \\ x_{random}, & P \geq P_{goal} \end{cases} \quad (2)$$

### 2) 搜索式连接引导路径

由于 RRT-Connect 算法在通道环境中获取有效采样点的概率较低,其在具有通道场景的环境中往往难以找到通道入口,并且难以在通道内完成路径规划。使用快速桥接法构建狭窄通道引导路径可以避免随机树在通道内部进行采样拓展,并且可以引导随机树快速搜索到通道入口,提高具有通道场景下的规划成功率和规划效率。

如图 11 所示,随机树每次获得新的节点  $x_{new}$  时在其周围一定范围内进行搜索,若是有引导节点位于其搜索范围内则通过该引导节点实现拓展树与引导路径的连接。引导路径与拓展树连接后成为拓展树的一部分,参与后续拓展树生长和搜索过程。算法持续迭代,直到两棵拓展树相遇,从相遇点向两棵拓展树回溯,得到从起点到终点的可通行路径 DLGS-RRT-Connect 算法的伪代码如算法 1 所示。

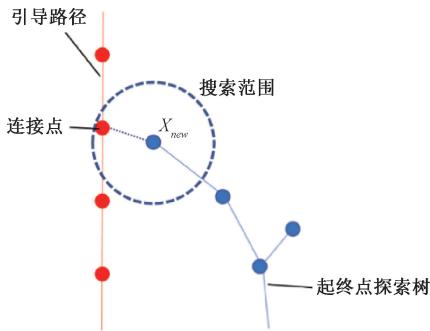


图 11 引导路径与拓展树连接

Fig. 11 Guided paths and expansion tree connections

## 算法 1: DLGS-RRT-Connect 算法

```

输入: 起点  $x_{init}$ , 目标点  $x_{goal}$ , 障碍物  $obs$ , 拓展步长  $steep$ , 最大迭代次数  $K$ , 特征点集合  $key\_nodes$ 
输出: 拓展树  $T_1, T_2$ 
1:  $V_1 \leftarrow \{x_{init}\}$ ;  $E_1 \leftarrow \emptyset$ ;  $T_1 \leftarrow (V_1, E_1)$  ;
2:  $V_2 \leftarrow \{x_{goal}\}$ ;  $E_2 \leftarrow \emptyset$ ;  $T_2 \leftarrow (V_2, E_2)$ ;  $i \leftarrow 0$ ;
3: while  $i < K$  do
4:    $x_{rand1} \leftarrow \text{Sample}(i)$ ;  $i \leftarrow i + 1$ ;
5:    $x_{near1} \leftarrow \text{Nearst}(T_1, x_{rand1})$ ;
6:    $x_{new1} \leftarrow \text{Steer}(x_{near1}, x_{rand1})$ ;
7:   if not in_barrier( $x_{new1}$ ) and not collision( $x_{near1}, x_{new1}$ ) then
8:      $V_1 \leftarrow V_1 \cup \{x_{new1}\}$ ;
9:      $E_1 \leftarrow E_1 \cup \{(x_{near1}, x_{new1})\}$ ;
10:     $x_{near2} \leftarrow \text{Nearst}(T_2, x_{new1})$ ;
11:     $x_{new2} \leftarrow \text{Steer}(x_{near2}, x_{new1})$ ;
12:    for node in  $key\_nodes$  do
13:      if near( $x_{new1}$ , node) then
14:         $V_1 \leftarrow V_1 \cup \{node\}$ ;
15:        for node in  $key\_nodes$  do
16:          if connect(node, node1) then
17:             $V_1 \leftarrow V_1 \cup \{node_1\}$ ;
18:          if not in_barrier( $x_{new2}$ ) and not collision( $x_{near2}, x_{new2}$ ) then
19:             $V_2 \leftarrow V_2 \cup \{x_{new2}\}$ ;
20:             $E_2 \leftarrow E_2 \cup \{(x_{near2}, x_{new2})\}$ ;
21:            do
22:               $x'_{new2} \leftarrow \text{Steer}(x_{near2}, x_{new2})$ ;
23:              if not in_barrier( $x'_{new2}$ ) and not collision( $x'_{near2}, x'_{new2}$ ) then
24:                 $V_2 \leftarrow V_2 \cup \{x'_{new2}\}$ ;
25:                 $E_2 \leftarrow E_2 \cup \{(x'_{new2}, x_{new2})\}$ ;
26:                 $x_{new2} \leftarrow x'_{new2}$ ;
27:              else break;
28:            while not  $x_{new2} = x_{new1}$ 
29:            if  $x_{new2} = x_{new1}$  then return  $(V_1, E_1), (V_2, E_2)$ ;
30:            if  $|V_2| < |V_1|$  then Swap( $V_1, V_2$ );
31: return  $T_1, T_2$ 

```

如图 12 (a) 所示, 在需要通过狭窄通道的环境中, RRT-Connect 算法的起点树与终点树都在各自的空间中进行了大量采样, 但一直达到迭代次数上限也未必能成功寻找到可行驶路径。这也反映了 RRT-Connect 算法在

需要穿越狭窄通道的场景中往往难以获得有效采样点, 从而导致其有较高可能无法寻找到通道入口并在狭窄通道中规划出安全的可行驶路径。如图 12 (b) 所示, DLGS-RRT-Connect 算法通过将起点树或终点树与引导路径进行连接, 使得拓展树能够快速生长通过狭窄通道, 解决了采样算法难以在狭窄通道中形成路径的问题。

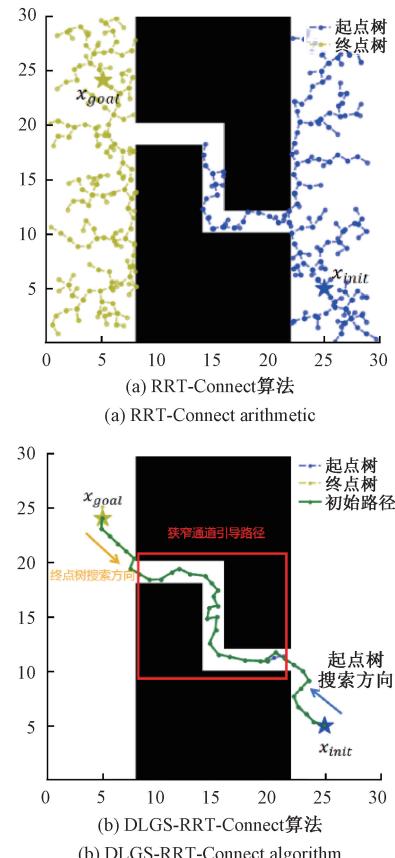


图 12 改进前后的 RRT-Connect 算法

Fig. 12 RRT-Connect algorithm before and after improvement

## 2.3 路径优化

无人车辆的运动需满足其运动学约束, 路径应平滑, 并尽量减少转弯次数和转弯半径。尽管 DLGS-RRT-Connect 算法能较好地规划狭窄通道内的路径, 但由于采样拓展和通道引导点生成的随机性, 规划路径中存在冗余点, 导致路径曲折蜿蜒, 不符合无人车辆的运动需求。为此, 本文采用剪枝的方法去除冗余点, 结合转角优化提升规划路径在通道转角处的安全性, 并利用四阶贝塞尔曲线对路径进行平滑处理。

## 1) 路径关键点提取

如图 13 (a) 所示, 从路径点  $x_1$  开始对路径进行剪枝, 连接点  $x_1$  和  $x_3$ , 若是两点的连线不与障碍物碰撞则连接  $x_1$  和  $x_3$ , 如此剔除冗余点  $x_2$ 。由于三角形两边之和大于第三边, 剪枝后的路径必然小于原有路径。如图 13 (b) 所示, 当对整条路径进行剪枝时, 从起点  $x_{init}$  开始

依次连接路径上后续节点进行剪枝操作,直到两点连线与障碍物碰撞。接着以最后一次剪枝操作的连线节点  $x_1$  为剪枝起点进行剪枝操作。以此类推,直到完成整条路径的剪枝优化。如图 13 (c) 所示,经过剪枝处理后的规划路径的质量有显著提升。

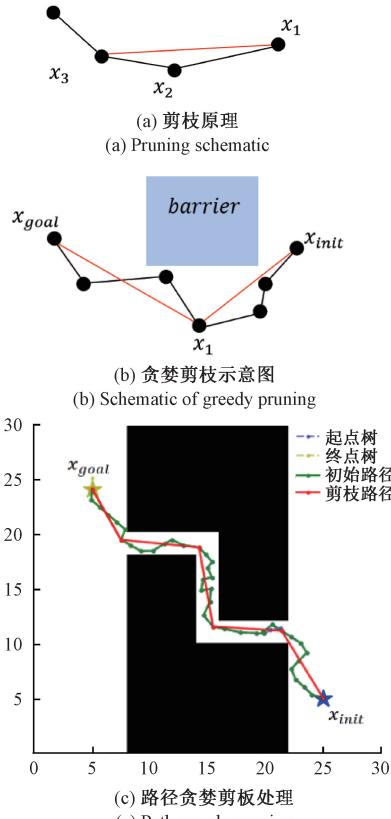


图 13 路径剪枝处理  
Fig. 13 Route pruning

## 2) 转角优化

剪枝后的路径在狭窄通道的转弯处可能过于靠近障碍物,在拟合平滑后易与障碍物发生碰撞。此外,由于无人车辆具有一定体积,路径在拐弯处需与障碍物保持足够的安全距离。因此,将剪枝路径的转角节点沿角平分线方向向外贪婪扩张一定距离,在保证剪枝路径安全性的前提下使其远离转弯方向。

转角扩张后,剪枝路径在转弯处的曲率增大,可能超出无人车辆底盘的最大转向角限制。为此,本文采用转角剪切的方法,对扩张后的转角进行角度约束,使平滑后的路径更贴合剪枝路径,并降低碰撞风险。

如图 14 所示,进行转角优化后的规划路径更加符合无人车辆在狭窄通道中的行驶需求。

## 3) 路径平滑

剪枝路径在转角处二阶不连续,为使路径更加符合车辆实际运动的需求,本文使用四阶贝塞尔曲线对剪枝

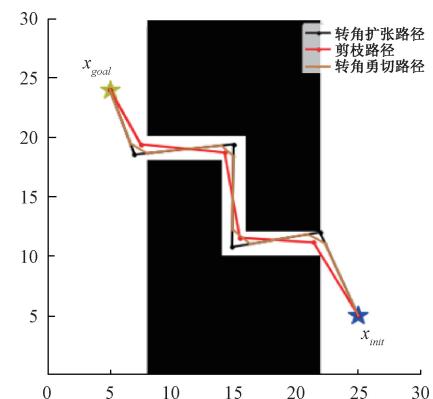


图 14 剪枝路径转角优化

Fig. 14 Pruning path corner optimization

路径进行平滑。 $n$  阶贝塞尔曲线的公式如式(3)所示。

$$\begin{cases} C(u) = \sum_{i=0}^n B_{n,i}(u) P_i \\ B_{n,i}(u) = \frac{n!}{i! (n-i)!} u^i (1-u)^{n-i} \end{cases} \quad (3)$$

式中: $P_i$  为贝塞尔曲线的控制点,这些控制点确定了贝塞尔曲线的形态; $\mu$  代表贝塞尔曲线的参数; $B_{n,i}$  为  $n$  次 Bernstein 基函数。

通过在路径上确认 5 个控制点即可构造一个四阶贝塞尔曲线对路径进行拟合。为了保证拟合效果,本文在剪枝路径中插入辅助点,采用分段拟合的方式实现对规划路径的平滑。为了保证平滑路径的二阶连续性,每一段拟合曲线的起点与前一段拟合曲线的终点相重合。插值点的分布如图 15 所示,其中  $x_i$  ( $i=1,2,3,4$ ) 为剪枝路径的节点,每两个剪枝节点之间插入 3 个辅助点  $P_{i,j}$  ( $i=1,2; j=1,2,3,4,5$ )。以剪枝节点为中心左右分别取两个辅助点构成一组控制点,基于控制点构造四阶贝塞尔曲线对折线  $P_{i,1}P_{i,5}$  进行拟合,实现该段路径的平滑。将多段平滑路径进行组合拼接即可实现对整段剪枝路径的平滑,平滑路径如图 16 所示。

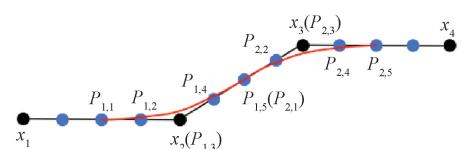


图 15 分段平滑

Fig. 15 Segmental smoothing

## 3 测试结果与讨论

### 3.1 仿真实验环境

地下管网、矿井隧道等场景通常由错综复杂的狭窄通道结构组成,并且因功能需求或自然成因形成大量厅

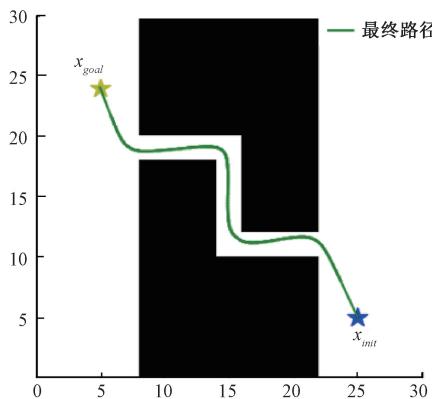
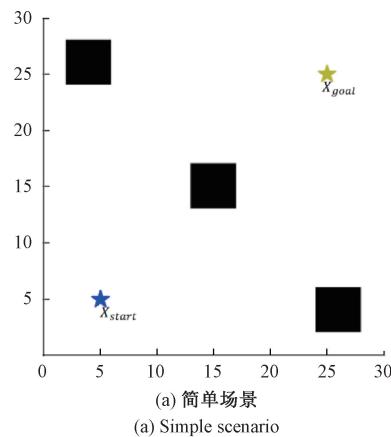
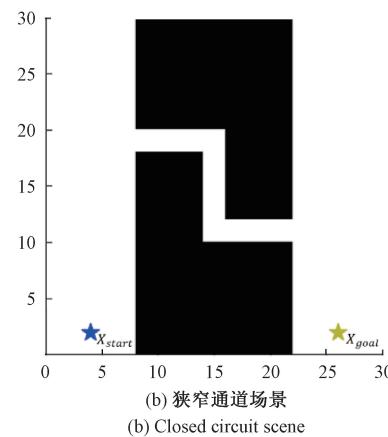


图 16 规划路径平滑处理

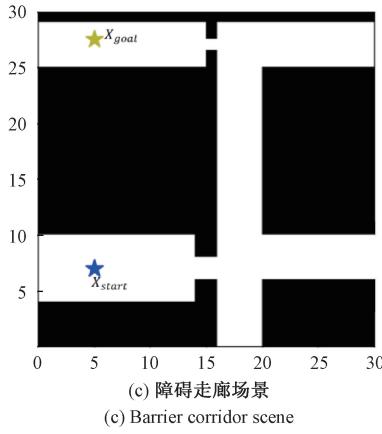
Fig. 16 Planning path smoothing



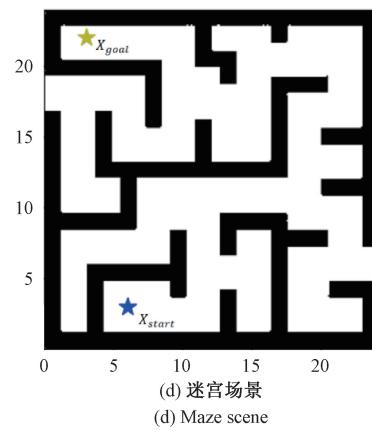
(a) 简单场景



(b) 狭窄通道场景



(c) 障碍走廊场景



(d) 迷宫场景

图 17 仿真环境地图

Fig. 17 Simulation environment map

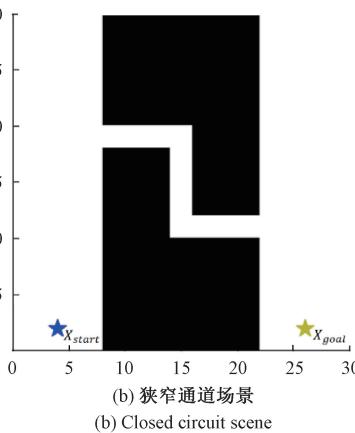
### 3.2 简单场景

在障碍物稀疏的简单场景下 Goal-bias RRT、Informed-RRT\*、RRT-Connect 以及 DLGS-RRT-Connect 四种路径规划算法的规划结果如图 18 所示。表 1 为 4 种算法在简单场景中的规划性能指标进行了统计和对比。

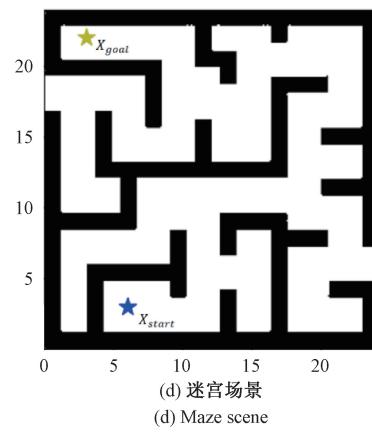
式联通空间,致使可通行区域拓扑结构高度复杂。同时,多岔路的迷宫特性进一步增加了无人系统的路径规划难度。

为验证 DLGS-RRT-Connect 算法在此类狭窄复杂空间中进行路径规划的性能,本文针对性地设计了狭窄通道场景、障碍走廊场景以及迷宫场景。此外,为了检验算法在较为常见的开阔环境中的规划性能,测试算法的环境适应性,本文还设计了一种障碍物稀疏的简单场景。仿真场景如图 17 所示。

本文利用 Python 语言编写程序,在 4 种地图中进行了 Goal\_bias RRT、Informed-RRT\*、RRT-Connect、DLGS-RRT-Connect 算法的仿真对比实验。将每种算法在各场景中重复实验 100 次,统计并分析规划成功率、平均规划时长、平均迭代次数以及规划路径的平均长度。



(b) 狹窄通道场景



(d) 迷宫场景

由图 18(a)可知,Goal-bias RRT 算法通过引入目标导向策略,虽然在采样过程中提高了规划效率,但当随机树在沿终点生长的方向遇到障碍物时,缺乏有效的绕行机制,导致产生较多冗余采样点,从而使得规划路径呈现出较大的曲折性。图 18(b)为 Informed-RRT\* 算法的表现,该算法通过逐步收敛椭圆形采样区域来优化路径,使

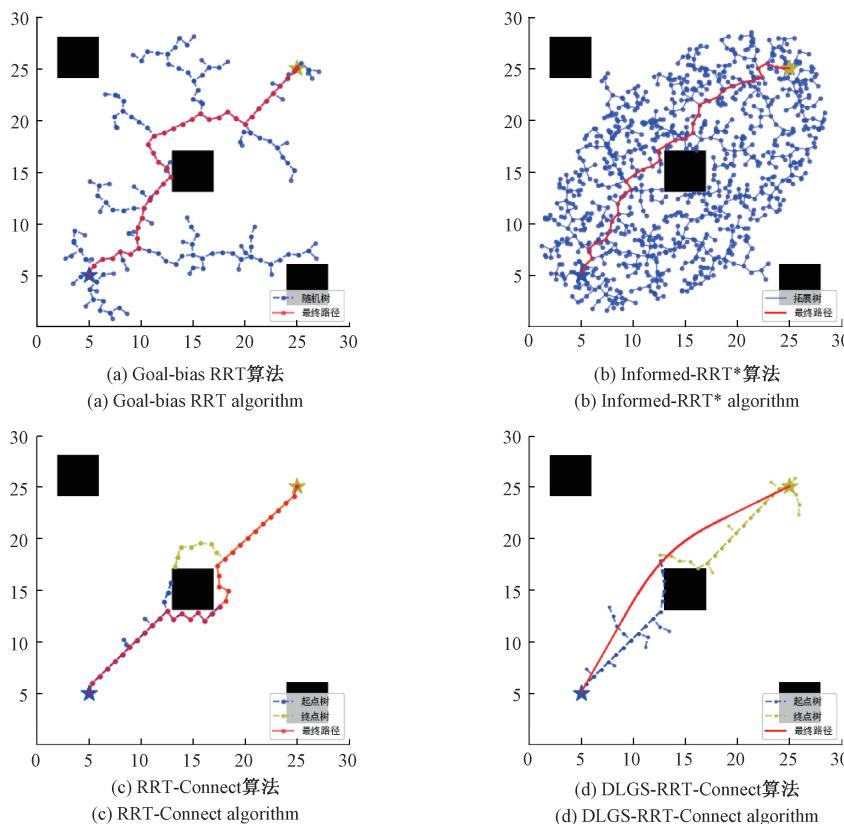


图 18 简单场景中各算法的路径规划结果

Fig. 18 Path planning results for each algorithm in a simple scenario

表 1 简单场景中各算法实验结果数据

Table 1 Experimental results data for each algorithm in simple scenarios

比较指标	Goal_bias RRT	Informed_RRT <sup>*</sup>	RRT_Connect	DLGS-RRT-Connect
规划成功率/%	100	100	100	100
平均规划时长/s	0.002 6	0.695 0	0.001 9	0.001 8
平均迭代次数	120	117	29	27
平均长度	34.913 7	34.787 0	33.509 2	29.129 8

得路径逐渐逼近最优解。然而,其迭代优化过程较为缓慢,带来了较大的时间开销,从而降低了整体规划效率。由图 18(c)可知,RRT-Connect 算法通过从起点和终点同时扩展两棵随机树,并采用贪婪策略促使树之间朝向彼此生长,显著提高了搜索效率。然而,其得到的路径质量较差,规划路径往往过于贴合障碍物,存在安全隐患。相比之下,DLGS-RRT-Connect 算法通过结合目标引导采样策略和路径后处理方法,能够在较短时间内高效规划出一条平滑且安全的路径,显著提高了路径规划的质量和效率。

由表 1 可知,在简单场景下,DLGS-RRT-Connect 算法相较于 Goal-bias RRT、Informed-RRT<sup>\*</sup> 和 RRT-Connect 算法,在规划时间上分别减少了 30.76%、99.74% 和 5.26%;在规划路径的平均长度上,分别减少了 16.57%、

16.26% 和 13.07%。这些结果表明,DLGS-RRT-Connect 在非狭窄通道的场景中能够提高规划效率,且在路径质量上也表现出较为明显的改善。

### 3.3 狹窄通道场景

在地下管网、矿井等场景中常通过狭窄通道连接各个区域。但是对于采样算法而言,如何在较为宽阔的区域中寻找较小的通道入口并在狭窄通道中快速规划出平滑且安全的可通行路径是一个难题。为了验证各算法寻找通道入口并在狭窄通道内进行路径规划的能力,本文设计了一个狭窄通道场景。各算法在该场景下的路径规划结果如图 19 所示,表 2 为 4 种算法的规划性能指标对比分析。

由图 19(a)、(b)可知,Goal-bias RRT 及 Informed-RRT<sup>\*</sup> 算法需要在非目标区域中进行较多采样才能够找

到狭窄通道入口,增添了规划过程的时间消耗和迭代次数。除此以外,由于狭窄通道中难以通过随机采样获得有效采样点,导致 Informed-RRT\* 算法在的规划成功率较低,仅为 48%。并且当地形限制椭圆采样区域难以缩小时,Informed-RRT\* 算法的规划路径难以收敛到最优。由图 19(a)、(c) 可知,虽然 Goal-bias RRT 和 RRT-Connect 算法分别通过目标导向策略和贪婪策略引导采样点的生成,提高了规划成功率,但其对随机树生长方向的引导也导致了规划路径的质量较低,具有贴近障碍物和存在较多波折的现象。由图 19(d) 可知,DLGS-RRT-Connect 能够通过搜索式连接策略以较少次采样找到通道入口,并通过连接狭窄通道引导路径快速规划出可行驶路径。同

时,由于狭窄通道桥梁点聚类之后多位于通道中线附近,DLGS-RRT-Connect 规划出的路径在安全性方面有所提升。

由表 2 可知,在狭窄通道场景中,DLGS-RRT-Connect 相较于另外 3 种算法,在平均规划时长上分别减少了 48.23%、97.33%、47.48%,在平均迭代次数上分别减少了 92.32%、94.68%、84.25%,在路径的平均长度上分别减少了 21.50%、25.50%、24.86%。这些指标的提升表明,DLGS-RRT-Connect 能够在狭窄通道场景中高效规划出安全且平滑的路径,体现了更强的计算性能和规划路径的优越性。

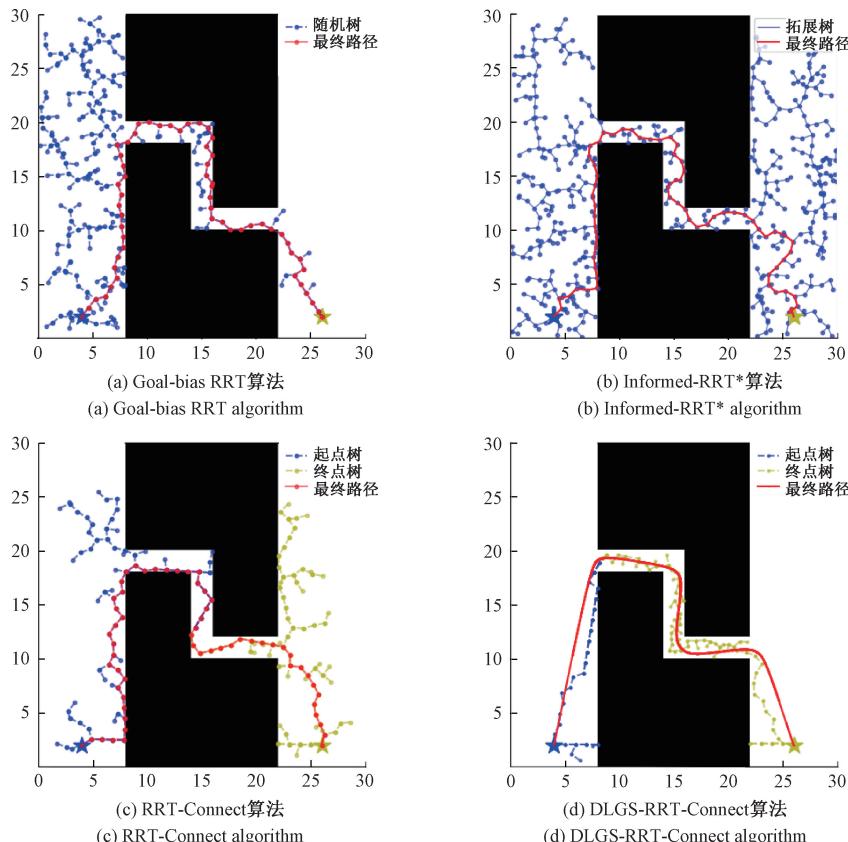


图 19 狹窄通道中各算法的路径规划结果

Fig. 19 Path planning results for each algorithm in a narrow channel

表 2 狹窄通道场景中各算法实验结果数据

Table 2 Experimental result data of each algorithm in narrow channel scene

比较指标	Goal_bias RRT	Informed_RRT <sup>*</sup>	RRT_Connect	DLGS-RRT-Connect
规划成功率/%	94	48	97	100
平均规划时长/s	0.0141	0.2733	0.0139	0.0073
平均迭代次数	482	696	235	37
平均长度	55.4601	58.4430	57.9409	46.5385

### 3.4 障碍走廊场景

走廊场景是地下管网、矿井隧道等室内环境中最为

常见的场景之一,无人系统常常需要从一个区域通过狭窄的门框和走廊岔路进入另一个区域。在这种环境中,

如何快速规划出既安全又平滑的可通行路径,是一个具有挑战性的任务。为验证算法在这种环境下的规划性能,本文设计了障碍走廊场景。在此场景中,无人车辆需要通过两个狭小的门框以及一段具有岔路的狭窄走廊方能成功抵达目标点。各算法在该场景下的路径规划结果如图 20 所示,表 3 为 4 种算法的规划性能指标对比分析。

由图 20 可知,Goal-bias RRT 和 Informed-RRT\* 算法依赖于大量采样才能生成规划路径,这不仅导致较高的计算开销,还导致规划路径较为曲折。同时,由于障碍走廊场景中的可通行区域占比较小,使得 Goal-bias RRT 和 Informed-RRT\* 算法获得有效采样点的效率较低,导致在障碍走廊场景中其规划成功率较低。RRT-Connect 算法虽通过双向搜索和贪婪策略提高了规划成功率,但其规

划路径会出现过于贴近障碍物以及大幅度转折的现象。由图 20(d)可知,DLGS-RRT-Connect 能够以极少次采样完成随机树与通道引导路径的连接,同时受益于目标引导采样策略和贪婪策略,实现了障碍走廊环境中的快速路径规划。

由表 3 可得,在障碍走廊场景中,相较于另外 3 种算法,DLGS-RRT-Connect 在规划成功率上分别提升了 35%、60%、26%,在平均规划时长上分别减少了 70.62%、97.65%、63.92%,在平均迭代次数上分别减少了 99.66%、99.73%、99.54%,在规划路径的平均长度上分别减少了 14.53%、16.70%、18.84%。实验结果表明,DLGS-RRT-Connect 在狭窄且多障碍的走廊通道场景中具有明显优势。它通过构建狭窄通道引导路径快速高效地规划出了安全且平滑的行驶路径。

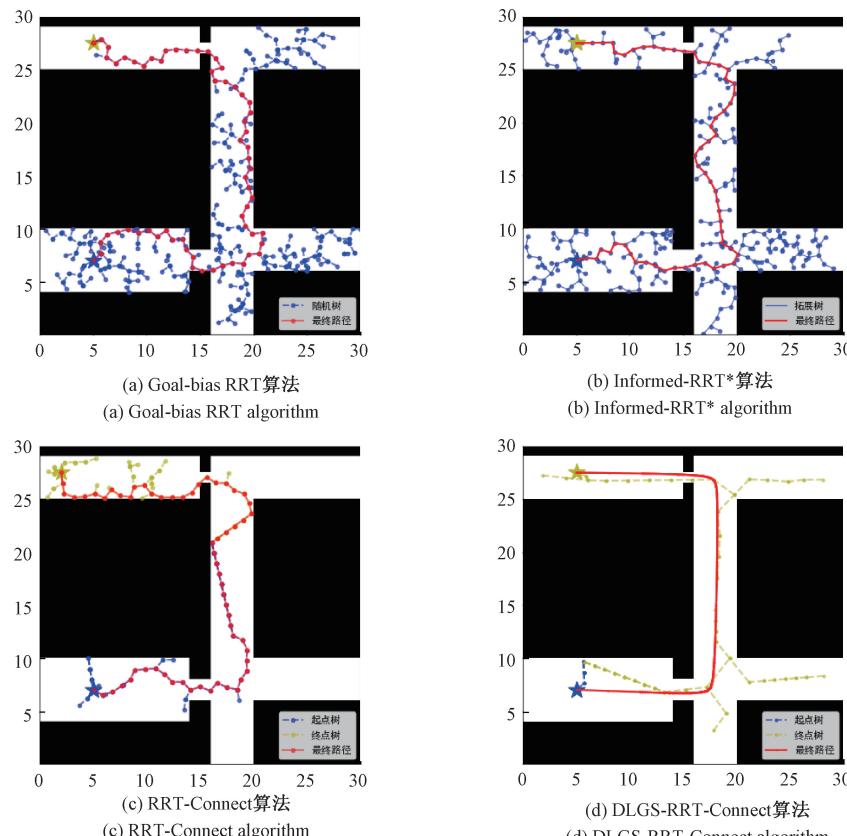


图 20 障碍走廊场景中各算法的路径规划结果

Fig. 20 Path planning results for each algorithm in the obstacle corridor scenario

表 3 障碍走廊场景中各算法实验结果数据

Table 3 Experimental result data for each algorithm in the obstacle corridor scenario

比较指标	Goal_bias RRT	Informed_RRT <sup>*</sup>	RRT_Connect	DLGS-RRT-Connect
规划成功率/%	65	40	74	100
平均规划时长/s	0.019 4	0.243 0	0.015 8	0.005 7
平均迭代次数	588	729	432	2
平均长度	53.080 3	54.460 2	55.902 7	45.368 0

### 3.5 迷宫场景

地下管廊、矿井等复杂场景通常由狭窄曲折的走廊组成,具有较为复杂的连通性和迷宫般的结构。如何在大量交错复杂的通道岔路中高效地规划出可行驶路径对于无人系统的路径规划来说是一个巨大的挑战。为了验证算法在多岔路狭窄复杂环境中的规划性能,本文设计了一种迷宫场景。各算法在该场景下的路径规划结果如图 21 所示,表 4 为 4 种算法的规划性能指标对比分析。

由图 21(a)、(b) 可知,在岔路较多的复杂迷宫场景中 Goal-bias RRT 和 Informed-RRT\* 算法需要在起始点周围进行大量采样才能靠近目标区域进行搜索,这种尝试性的采样效率极低,导致两种算法难以在岔路较多的迷宫场景中成功规划出路径。RRT-Connect 算法虽然通过双向拓展和贪婪策略提高了规划的成功率,但也因为这

种强引导性采样的方式导致其生成的规划路径易出现极端转折现象和过于贴近障碍物的现象,路径质量较差。由图 21(d) 可知,DLGS-RRT-Connect 能够通过连接狭窄通道,引导随机树迅速拓展至迷宫场景的各个通道,从而高效规划出安全且平滑的行驶路径。

由表 4 可得,在迷宫场景中,相较于另外 3 种算法,DLGS-RRT-Connect 在规划成功率上分别提升了 92%、93%、34%,在平均规划时长上分别减少了 21.11%、86.76%、24.11%,在平均迭代次数上分别减少了 99.78%、99.78%、99.71%,在规划路径的平均长度上分别减少了 18.44%、15.48%、21.18%。实验结果表明,DLGS-RRT-Connect 能够在复杂的迷宫场景中,快速高效地规划出高质量的路径,尤其适用于包含大量岔路和走廊的环境。



图 21 迷宫场景中各算法的路径规划结果

Fig. 21 Path planning results for each algorithm in the maze scenario

表 4 迷宫场景中各算法实验结果数据

Table 4 Experimental result data of each algorithm in the maze scenario

比较指标	Goal_bias RRT	Informed_RRT*	RRT_Connect	DLGS-RRT-Connect
规划成功率/%	8	7	66	100
平均规划时长/s	0.037 9	0.225 8	0.039 4	0.029 9
平均迭代次数	890	912	679	2
平均长度	58.207 4	56.168 5	60.232 8	47.474 7

在3种不同的具有狭窄通道的环境中,DLGS-RRT-Connect 算法相较于其他算法在规划成功率、平均规划时长、平均迭代次数以及规划路径的平均长度上均有显著优势。该算法通过引导路径指引随机树在狭窄通道中的拓展,极大地提升了在狭窄通道中的规划效率。此外,在普通场景中,该算法虽然没有展现出其独特的优势,但与其他算法相比性能仍有提升。同时,该算法通过路径优化处理显著提高了路径的平滑度和安全性,弥补了采样算法路径质量差的问题。总体而言,DLGS-RRT-Connect 算法在多种场景中均展现出了其高效、高质的路径规划能力。

## 4 结 论

针对 RRT 算法在狭窄复杂环境中易于出现的规划成功率低、耗时长以及规划路径质量差的问题,本文提出了 DLGS-RRT-Connect。该算法基于快速桥接法,利用栅格地图构建场景中狭窄通道的引导路径,从而实现对狭窄复杂场景的空间特征描述。通过狭窄通道引导路径辅助搜索过程,使得随机树在狭窄通道场景中快速高效拓展,从而改善了算法在狭窄通道中难以获得有效采样点的问题,提升了在狭窄复杂环境中的搜索效率。同时,在路径点的采样过程中,算法引入了目标偏置引导策略,降低了采样的随机性,进一步提高了规划速度。此外,通过贪婪剪枝和分段四阶贝塞尔平滑的方式对规划路径进行后处理,显著提高了规划路径的质量。

实验结果表明,在狭窄通道、障碍走廊以及迷宫场景中,相较于 Goal\_bias RRT、Informed-RRT\* 以及 RRT-Connect 算法,DLGS-RRT-Connect 的规划成功率最高,消耗时间最少,路径长度最短。尤其在狭窄且复杂的通道环境中,DLGS-RRT-Connect 算法在规划成功率上相较于 Goal\_bias RRT、Informed-RRT\* 和 RRT-Connect 算法分别提高了 35%、60%、26%。在平均规划时长上,相较于以上算法分别降低了 70.62%、97.65%、63.92%。规划路径的平均长度也分别减少了 14.53%、16.70%、18.84%。体现出了该算法在狭窄复杂空间中进行路径规划的显著优势。此外,在简单场景下,该算法在规划效率和规划路径的质量上都有显著提升。总体而言,DLGS-RRT-Connect 算法具有较高的环境适用性。

## 参考文献

- [ 1 ] SZREK J, ZIMROZ R, WODECKI J, et al. Application of the infrared thermography and unmanned ground vehicle for rescue action support in underground mine—The amicos project [ J ]. Remote Sensing, 2020, 13( 1 ): 69.
- [ 2 ] GUO P, SUN C, LI Q. Obstacle avoidance path planning in unstructured environment with narrow passages [ J ]. IEEE Transactions on Intelligent Vehicles, 2023, 8( 11 ): 4632-4643.
- [ 3 ] 肖金壮,余雪乐,周刚,等.一种面向室内 AGV 路径规划的改进蚁群算法 [ J ]. 仪器仪表学报, 2022, 43( 3 ): 277-285.
- XIAO J ZH, YU X L, ZHOU G, et al. An improved ant colony algorithm for indoor AGV path planning [ J ]. Chinese Journal of Scientific Instrument, 2022, 43( 3 ): 277-285.
- [ 4 ] LIU L, WANG X, YANG X, et al. Path planning techniques for mobile robots: Review and prospect [ J ]. Expert Systems with Applications, 2023, 227: 120254.
- [ 5 ] TENG S, HU X, DENG P, et al. Motion planning for autonomous driving: The state of the art and future perspectives [ J ]. IEEE Transactions on Intelligent Vehicles, 2023, 8( 6 ): 3692-3711.
- [ 6 ] 张俊豪,潘树国,高旺,等.基于改进 RRT 算法的狭长空间无人车辆路径规划 [ J ]. 全球定位系统, 2023, 48( 4 ): 81-90.
- ZHANG J H, PAN SH G, GAO W, et al. Path planning of unmanned vehicles in narrow and long space based on improved RRT algorithm [ J ]. GNSS World of China, 2023, 48( 4 ): 81-90.
- [ 7 ] 孙凌宇,王威,秦红亮,等.跳点优化蚁群算法的移动机器人路径规划 [ J ]. 电子测量技术, 2023, 46( 9 ): 48-53.
- SUN L Y, WANG W, QIN H L, et al. Mobile robot path planning based on jump point optimization ant colony algorithm [ J ]. Electronic Measurement Technology, 2023, 46( 9 ): 48-53.
- [ 8 ] 崔炜,朱发证.机器人导航的路径规划算法研究综述 [ J ]. 计算机工程与应用, 2023, 59 ( 19 ): 10-20.
- CUI W, ZHU F ZH, et al. Review of Path Planning Algorithms for Robot Navigation [ J ]. Computer Engineering and Applications, 2023, 59( 19 ): 10-20.
- [ 9 ] MIAO C, CHEN G, YAN C, et al. Path planning optimization of indoor mobile robot based on adaptive ant colony algorithm [ J ]. Computers & Industrial Engineering, 2021, 156: 107230.
- [ 10 ] 康亮,赵春霞,郭剑辉.基于模糊滚动 RRT 算法的移动机器人路径规划 [ J ]. 南京理工大学学报(自然科学版), 2010, 34( 5 ): 642-648.
- KANG L, ZHAO CH X, GUO J H, et al. Path planning based on fuzzy rolling rapidly-exploring random tree for mobile robot [ J ]. Journal of Nanjing University of Science and Technology, 2010, 34( 5 ): 642-648.
- [ 11 ] 许瑶,刘晓峰.基于改进 RRT 算法的无人车路径规划

- 研究与测试 [J]. 国外电子测量技术, 2023, 42(8): 132-138.
- XU Y, LIU X F, et al. Research and testing of unmanned vehicle path planning based on improved RRT algorithm [J]. Foreign Electronic Measurement Technology, 2023, 42(8): 132-138.
- [12] 崔锡杰, 王晓军, 李晓航. 改进 RRT 算法的机器人全局路径规划 [J]. 计算机工程与应用, 2025, 61(4): 331-338.
- CUI X J, WANG X J, LI X H, et al. Robot global path planning based on improved RRT algorithm [J]. Computer Engineering and Applications, 2025, 61(4): 331-338.
- [13] CHI W, DING Z, WANG J, et al. A generalized Voronoi diagram-based efficient heuristic path planning method for RRTs in mobile robots [J]. IEEE Transactions on Industrial Electronics, 2021, 69(5): 4926-4937.
- [14] 王萍, 潘树国, 蔚保国, 等. 基于 RSS\_GN RRT 算法的狭长空间路径规划 [J]. 电子测量与仪器学报, 2024, 38(1): 72-85.
- WANG P, PAN S G, YU B G, et al. Narrow and long space path planning based on RSS\_GN RRT algorithm [J]. Journal of Electronic Measurement and Instrumentation, 2024, 38(1): 72-85.
- [15] KUFFNER J J, LAVALLE S M. RRT-connect: An efficient approach to single-query path planning [C]. Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. IEEE, 2000, 2: 995-1001.
- [16] LAVALLE S M, KUFFNER JR J J. Randomized kinodynamic planning [J]. The International Journal of Robotics Research, 2001, 20(5): 378-400.
- [17] KARAMAN S, FRAZZOLI E. Sampling-based algorithms for optimal motion planning [J]. The International Journal of Robotics Research, 2011, 30(7): 846-894.
- [18] GAMMELL J D, SRINIVASA S S, BARFOOT T D. Informed RRT\*: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic [C]. 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 2014: 2997-3004.
- [19] LIAO B, WAN F Y, HUA Y, et al. F-RRT\*: An improved path planning algorithm with improved initial solution and convergence rate [J]. Expert Systems With Applications, 2021, 184: 115457.
- [20] LI Y, WEI W, GAO Y, et al. PQ-RRT\*: An improved path planning algorithm for mobile robots [J]. Expert Systems with Applications, 2020, 152: 113425.
- [21] 李耀东, 苗春艳, 高健, 等. 基于改进 Prim 算法的路径规划研究 [J]. 现代电子技术, 2024, 47(4): 176-181.
- LI Y D, MIAO CH Y, GAO J, et al. Research on path planning based on improved Prim algorithm [J]. Modern Electronics Technique, 2024, 47(4): 176-181.

### 作者简介



薛佳慧, 2024 年于南京理工大学获得学士学位, 现为东南大学硕士研究生, 主要研究方向为无人车辆路径规划。

E-mail: xjh2206561185@163.com



Xue Jiahui received her B. Sc. degree from Nanjing University of Science and Technology in 2024. Now she is a M. Sc. candidate of Southeast University. Her main research interest includes unmanned vehicle path planning.

陶贤露(通信作者), 2023 年于武汉大学获得博士学位, 现为东南大学讲师, 硕士生导师, 主要研究方向为北斗/GNSS 高精度定位与导航场景智能感知。

E-mail: xltao@seu.edu.cn

Tao Xianlu (Corresponding author), received his Ph. D. degree from Wuhan University in 2023. He is now a lecturer and master's supervisor at Southeast University. His main research interests include BeiDou/GNSS high-precision positioning and intelligent perception of navigation scenes.