

DOI: 10.13382/j.jemi.B2306832

基于 RSS_GN RRT 算法的狭长空间路径规划*

王萍¹ 潘树国¹ 蔚保国² 高旺¹ 胡鹏¹

(1. 东南大学仪器科学与工程学院 南京 210096; 2. 卫星导航系统与装备技术国家重点实验室 石家庄 050081)

摘要:针对 RRT 算法在狭长空间中存在的收敛速度慢及规划路径不平滑的问题,提出了一种 RSS_GN RRT 算法。为了提升算法的收敛速度,提出了引导节点导向策略与分区域采样策略,极大地减少了算法对无效区域的搜索;其次,算法引入了采样角度约束策略来提高规划路径的质量,并采用父节点拓展选择的方法有效解决了由角度约束引起的迭代次数增加的问题。此外,算法可根据感知信息进行地图动态重构并规划避障路径,提高了算法在低速动态环境中的适应性。仿真结果显示,在狭长通道环境中,RSS_GN RRT 算法在规划路径的耗时上比 RRT、Goal_bias RRT、角度约束下的 RRT、Informed RRT* 及 DR-RRT 算法分别减少了 77.3%、51.9%、84.7%、98.8% 和 60.3%。在迭代次数上,相比于上述算法,分别减少了 95.9%、92%、98.3%、98.3% 和 89.5%。路径的平均曲率也分别降低了 94.1%、93.2%、88.7%、91% 和 92.9%。仿真结果证明了 RSS_GN RRT 算法在提升规划速度和改善路径质量方面具有显著优势。同时,本文采用了阿克曼模型的小车实测了算法的局部避障能力。经测试,小车可对行驶途中出现的障碍物进行合理避让。

关键词: RSS_GN RRT; RRT; 引导节点; 分区域采样; 角度约束; 地图重构

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 long space path planning based on RSS_GN RRT algorithm

Wang Ping¹ Pan Shuguo¹ Yu Baoguo² Gao Wang¹ Hu Peng¹

(1. School of Instrument Science and Engineering, Southeast University, Nanjing 210096, China;

2. State Key Laboratory of Satellite Navigation System and Equipment Technology, Shijiazhuang 050081, China)

Abstract: Aiming at the problems of RRT algorithm in narrow and long space, including slow convergence speed and rough planned path, the RSS_GN RRT algorithm was proposed. To enhance the algorithm's convergence speed, a guide node-oriented strategy and a regional sampling strategy was proposed, greatly reducing the search for invalid regions. Next, the sampling angle constraint strategy was introduced to improve the planned path quality, and adopted the method of parent node expansion selection to effectively solve the problem of increased iteration times caused by angle constraint. Furthermore, the algorithm can dynamically reconstruct the map and plan obstacle avoidance path based on the perception information, enhancing its adaptability in the low-speed dynamic environment. The simulation results show that in a narrow and long channel environment, the RSS_GN RRT algorithm reduces path planning time by 77.3%, 51.9%, 84.7%, 98.8%, and 60.3% when compared to the RRT, Goal_bias RRT, RRT under angle constraint, Informed RRT*, and DR-RRT algorithms, respectively. It decreases the number of iterations by 95.9%, 92%, 98.3%, 98.3%, and 89.5% relative to above algorithms. The average curvature of the path is also reduced by 94.1%, 93.2%, 88.7%, 91%, and 92.9%, respectively. The simulation results prove that RSS_GN RRT algorithm's significant advantages in enhancing planning speed and optimizing path quality. Simultaneously, this paper uses the Ackerman model car to actually measure the local obstacle avoidance ability of the algorithm. After testing, the car can reasonably avoid obstacles that appear during driving.

Keywords: RSS_GN RRT; RRT; guide nodes; region-state sampling; angle constraint; map reconstruction

0 引言

随着现代化物流系统的快速发展,智能无人车辆技术在地下物流领域的应用得到了广泛关注和研究。智能无人车辆系统利用导航定位、场景感知及运动控制技术^[1],使车辆能够在地下环境自主地执行搬运任务,提高了生产效率和资源的利用率。路径规划可提供一条自起始点至目标点的无碰撞安全路径,是智能无人车辆系统的关键环节。常用的全局路径规划算法有 A* 算法^[2]、RRT 算法(rapidly-exploring random tree)^[3]、蚁群算法^[4]、遗传算法^[5]等。RRT 算法采样速度快且具有概率完备性,可用于解决高维空间的路径规划问题。相比于其他算法,RRT 算法具有结构简单、搜索能力强的特点^[6]。但是,由于 RRT 算法的随机性与快速探索性,随机树的生长往往偏向未被探索的空间。特别是在封闭狭长的环境下,算法的收敛速度较慢,生成的路径也会含有许多不必要的转折点,增加了车辆的能量消耗以及控制的难度。

针对 RRT 算法存在的局限性,国内外学者提出了多种改进措施。文献[7]提出了一种改进的双向 RRT 算法,缩短了路径规划所需的时间。该算法以起点和终点之间的中点为根节点,同步构建两棵随机树进行扩展。在新节点扩展时,还引入了一种极度贪心的策略,用于判断是否可直接连通至终点。文献[8]提出了 NCC-RRT* 算法,在路径点采样时使用碰撞代价函数来替代传统的碰撞检测。这一改进减少了算法在碰撞检测时消耗的时间,提高了规划效率。文献[9]提出的 Informed RRT* 算法在找到初始路径后,构建了一个可逐渐收敛的椭圆,并将采样点限制在此椭圆中,通过不断迭代优化来提升路径质量。文献[10]提出的 RRT* Smart 算法利用智能采样策略来优化 RRT* 生成的初始路径,加快了 RRT* 算法的收敛速度。Informed RRT* 和 RRT* Smart 等算法都需要先通过 RRT* 算法搜索一条初始引导路径,再根据初始路径确定采样区域。在复杂的场景下,RRT* 的初始搜索效率不高。此外,为了减小 RRT 算法的随机性,文献[11]首次提出了目标点导向的方法,文献[12-14]均采用了这一思想实现了目标偏置的 RRT 算法(Goal_bias RRT)。其利用目标点作为采样点来引导随机树的生长,但该方法在选取目标点时的概率是固定的,在障碍物区域由终点引导生成的采样点通常是无效的。文献[15]在 Goal-bias RRT 算法的基础上提出了目标动态概率采样的方法。该方法根据障碍物的数量自适应地调整目标采样概率,增加了算法的场景适用性。文献[16-17]提出了基于人工势场引导下的改进 RRT 算法,通过建立人工势场来引导采样点远离障碍物区域,并向目标区域拓展,从而减小算法的随机性。文献[18]中提出的 DR-RRT

算法利用地图骨架来指导路径点的采样,使其沿骨架路径逐步进行采样。这种方法为路径点采样提供了明确的方向指引,加速了算法的收敛速度。

针对封闭狭长空间中 RRT 算法存在的规划效率低和路径质量差的问题,本文在 Goal_bias RRT 算法的基础上提出了一种 RSS_GN RRT(RRT algorithm with region-state sampling and guided by guide nodes)算法。首先,提出了引导节点导向策略。利用先验地图提取引导节点以引导随机树地扩展,为采样过程提供正确的方向指引,减小搜索的随机性与盲目性。其次,加入了分区域采样策略。根据区域状态自适应地调整采样方式,提高算法在空闲区域的收敛速度;与此同时,对采样点进行了角度约束,以提高路径质量。最后,为了增加算法在低速动态场景中的适用性,本文还利用场景感知系统提供的环境信息进行地图地实时重构与局部避障规划。

1 RRT 算法及改进

1.1 环境模型

已知的环境信息是路径规划的前提。本文采用二维栅格地图作为输入,为路径规划提供环境信息。其将环境分为可通行区域和不可通行区域的栅格单元,以数组形式进行存储,易于建模、更新和扩展。

1.2 RRT 算法

RRT 算法是一种基于随机采样的路径规划算法,以起点为随机树的根节点,通过随机采样将随机树扩展到可行驶的区域直至搜索到目标点,生成一条从起始点到目标点的全局规划路径。RRT 算法的路径点采样示意图如图 1 所示。

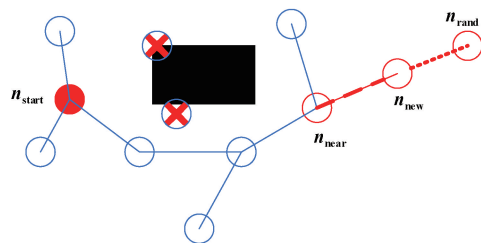


图1 路径点采样示意图

Fig. 1 Path point sampling diagram

RRT 算法的基本步骤如下:

- 1) 在可行驶的空间内生成随机采样点 n_{rand} 。
- 2) 计算随机树中距 n_{rand} 最近的节点 n_{near} , 以 n_{near} 为起点朝 n_{rand} 方向拓展一个步长, 生成新节点 n_{new} 。
- 3) 判断 n_{near} 和 n_{new} 构成的树枝是否发生碰撞。若不碰撞, 将 n_{new} 添加至随机树列表中; 反之, 重新生成 n_{rand} 。
- 4) 计算新节点与目标点的距离, 当距离小于 d 时, 完

成规划,回溯随机树,找到规划路径;反之,循环上述过程。

1.3 Goal_bias RRT 算法

Goal_bias RRT 算法在传统 RRT 算法的基础上提出了目标点导向策略,按一定概率 P 选择目标点作为随机采样点,引导随机树向目标点方向生长,在一定程度上提高了搜索效率。但该算法未考虑环境信息,仅用目标点来引导随机树的生长,使得在狭窄区域或复杂场景中,目标点导向策略会失效,造成大量计算资源的浪费。如图 2 所示,由终点 n_{goal} 引导生成得 n_{new1} 和 n_{new2} 节点与障碍物发生碰撞,拓展点无效。

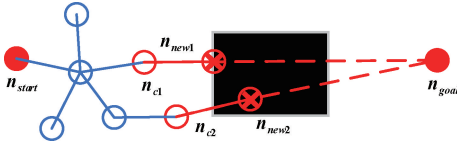


图 2 目标点导向策略失效示意图

Fig. 2 Guide node-oriented strategy failure diagram

2 RSS_GN RRT 算法

为了在狭长空间下快速生成合理且有效的路径,本文提出了一种改进的 RSS_GN RRT 算法。该算法加入了引导节点导向策略、角度约束策略以及分区域采样策略,提高了规划速度和路径质量;并对规划的路径进行了 5 次贝塞尔曲线平滑,以确保车辆运动的连续性与平稳性。

2.1 引导节点提取

本文提出的引导节点导向策略综合考虑了空间环境信息,根据先验的栅格地图及起始点位置提取出若干引导节点,以引导节点将规划分段,依次用引导节点引导采样点向目标区域拓展。根据地图提取的引导点能够准确地反映起终点间可行驶区域的位置与方向。当车辆位于狭窄道路或障碍物复杂区域时,引导节点的应用可有效避免算法对无效区域的搜索,减少无用的采样点,提高路径规划的效率。

引导节点的提取方式如下:

1) 提取骨架点。根据栅格地图提取地图的二值化骨架点。栅格地图如图 3(a) 所示,骨架提取结果如图 3(b) 所示。

2) 提取单一的骨架线,如图 3(c) 所示。根据起始点位置及环境信息提取出连通起终点的单一骨架线。

3) 提取引导节点。提取出单一骨架线中关键的转折点,并剔除距离过近的冗余转折点,最终得到引导节点,如图 3(d) 所示。

引导节点提取的伪代码如下:

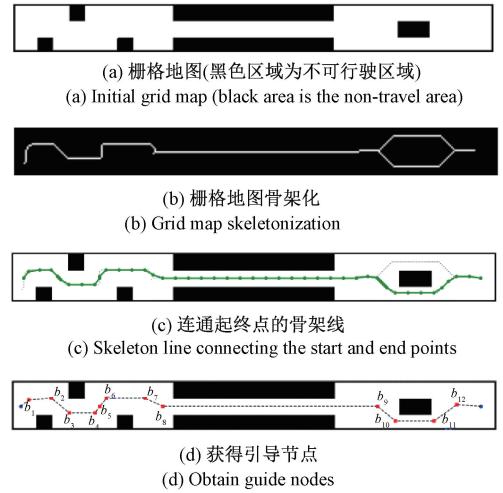


图 3 引导节点提取过程示意图

Fig. 3 Guide node extraction process diagram

算法 1: 引导节点提取算法

Input: x_{start} , x_{goal} , Map

Output: $KeyPoints$

```

1: function GNodeExtract ( $x_{start}$ ,  $x_{goal}$ ,  $Map$ )
2:    $SkelePoints \leftarrow \text{Skeletonize}(Map)$ 
3:   Add ( $SkelePoints$ ,  $x_{start}$ ,  $x_{goal}$ )
4:    $SkeleLine \leftarrow \text{GetLine}(SkelePoints, x_{start}, x_{goal})$ 
5:    $i = 1$ 
6:    $cNode \leftarrow SkeleLine[0]$ 
7:   while  $i < \text{len}(SkeleLine) - 1$  do
8:      $Node1 \leftarrow SkeleLine[i]$ 
9:      $Node2 \leftarrow SkeleLine[i+1]$ 
10:    if Judge( $cNode$ ,  $Node1$ ,  $Node2$ ) is false then
11:       $cNode \leftarrow SkeleLine[i]$ 
12:      Add ( $KeyPoints$ ,  $cNode$ )
13:    end if
14:     $i \leftarrow i + 1$ 
15:  end while
16:   $KeyPoints \leftarrow \text{DelExcessPoints}(KeyPoints)$ 
17: end function

```

2.2 角度约束

为了减小规划路径的曲折程度,本文引入了采样角度约束的策略^[19-20],即限制随机树中任意连续 3 点的夹角值小于约束角度 φ 。如图 4 所示,判断新节点 n_{new} 与父节点 n_{near} 间的角度 θ_1 及 n_{near} 与其父节点 n_{parent_odd} 的角度 θ_2 的差值 θ 是否满足约束角度,若不满足,则放弃此次拓展。然而,角度约束增加了新采样点的生成难度,导致算法的规划成功率下降。针对这一问题,文献[21]提出了一种限制区域随机转向的方法。该方法在采样点不满足角度限制时,会在限制范围内随机选择一个方向来生成新的采样点。这一方法在障碍物较少的空旷场景下优势明显,因为它在保证角度限制的同时增加了路径选

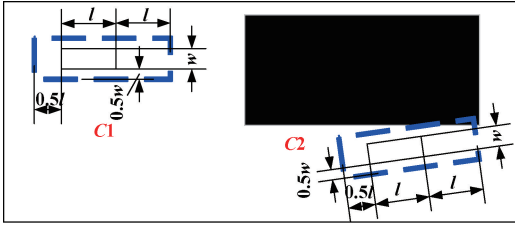


图 8 区域安全性判断示意图

Fig. 8 Regional security judgment diagram

路径点生成的具体流程如下：

- 1) 判断当前位置的区域状态,选择匹配的采样方式。
- 2) 生成新的采样点。若为引导采样的方式,直接生成新采样点,转至步骤 6);若使用随机采样的方式,随机生成采样点后对该点进行碰撞检测。
- 3) 碰撞检测。若新采样点产生了碰撞,则转至步骤 2),进入下一次随机采样;反之,对该点进行角度约束检测。
- 4) 角度约束检测。当新采样点满足角度约束,则转至步骤 6);反之,进行父节点拓展选择。
- 5) 父节点拓展选择。在待选的区域寻找是否存在满足角度约束的父节点,若不存在,则转至步骤 2),进入下一次随机采样;反之,进行目标点到达检测。
- 6) 目标点到达检测。若新采样点到达目标点,则完成规划;反之,转至步骤 1),继续迭代,直至到达目标点或达到最大的迭代次数。

路径点生成函数的伪代码如下：

算法 2: 路径点生成算法

Input: x_{start} , x_{goal} , $Grid_map$, $KeyPoints$

Output: $Pathpoints$

1: **function** PathSampling(x_{start} , x_{goal} , $Grid_map$, $KeyPoints$)

2: $cNode \leftarrow x_{start}$

3: **for** $i=0$ to $i=n$ **do**

4: **if** AreaState($cNode$) = true **then**

5: $x_{new} \leftarrow \text{GuideSample}(cNode)$

6: Add($Tree$, x_{new})

7: $cNode \leftarrow x_{new}$

8: **else**

9: $x_{new} \leftarrow \text{RandomSample}()$

10: **if** CollisionCheck(x_{new}) = false **then**

11: **if** AngleCheck(x_{new}) = true **then**

12: Add($Tree$, x_{new})

13: **else**

14: **if** FindNewParent(x_{new}) = true **then**

15: Add($Tree$, x_{new})

16: **else**

17: **continue**

18: **end if**

19: **end if**

20: **if** AreaState(x_{new}) = true **then**

```

21:     cNode ← xnew
22:   end if
23: end if
24: end if
25: if Distance(xnew, xgoal) < d then
26:   Pathpoints ← SearchPath(Tree)
27:   break
28: end if
29: end for
30: end function

```

2.4 路径平滑

本文采用五次贝塞尔曲线对离散的路径点进行分段平滑,为车辆提供曲率连续的路径。 n 阶贝塞尔曲线公式^[22]如式(3)所示。

$$\begin{cases} P(u) = \sum_{i=0}^n P_i B_{i,n}(u), u \in [0, 1] \\ B_{i,n}(u) = C_n^i u^i (1-u)^{n-i} \end{cases} \quad (3)$$

式中: P_i 为贝塞尔曲线的控制点; u 代表贝塞尔曲线的参数; $B_{i,n}$ 为 n 次 Bernstein 基函数。

每 6 个离散的路径点进行一次贝塞尔曲线拟合。为保证路径的一阶连续性,每一段曲线的起点与前一段曲线的终点重合,如图 9 中 P6 与 Q1 重合。此外,进行辅助点插值以保证分段处的曲率连续。辅助点插值的示意图如图 9, Q2 为插入的辅助点,并对 Q2 进行以下位置约束^[23]。

1) P6-P5 与 Q2-Q1 共线

2) $\|Q2-P5\| \geq \|P6-P5\| \geq \|Q2-Q1\|$

3) $\|Q2-Q1\| \leq \|Q3-Q2\|$

4) $Q2-Q1 = n(P6-P5)$, $n \in (0, 1]$

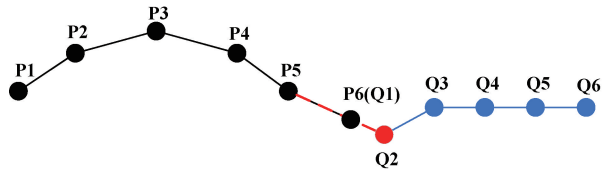


图 9 辅助点插值示意图

Fig. 9 Auxiliary point interpolation diagram

2.5 RSS_GN RRT 算法

本文提出的 RSS_GN RRT 算法首先需要获取先验栅格地图的骨架点,以此来确定引导节点。接着,算法在引导节点的指引下,根据车辆所处区域的状态自适应地选择路径点的采样方式。若为引导节点引导的采样方式,路径点会朝向引导节点的方向生成,而对于随机采样的方式,算法会随机生成路径点,再对其进行碰撞检测和角度限制检测。对于不满足角度限制的采样点,算法会扩

展选择其他潜在的父节点,并重新进行角度限制检测。这一过程将持续迭代,直至采样点到达目标位置。最后,算法将回溯并生成初始路径,再通过五次贝塞尔平滑处理得到一个平滑的路径。算法的流程如图 10 所示。

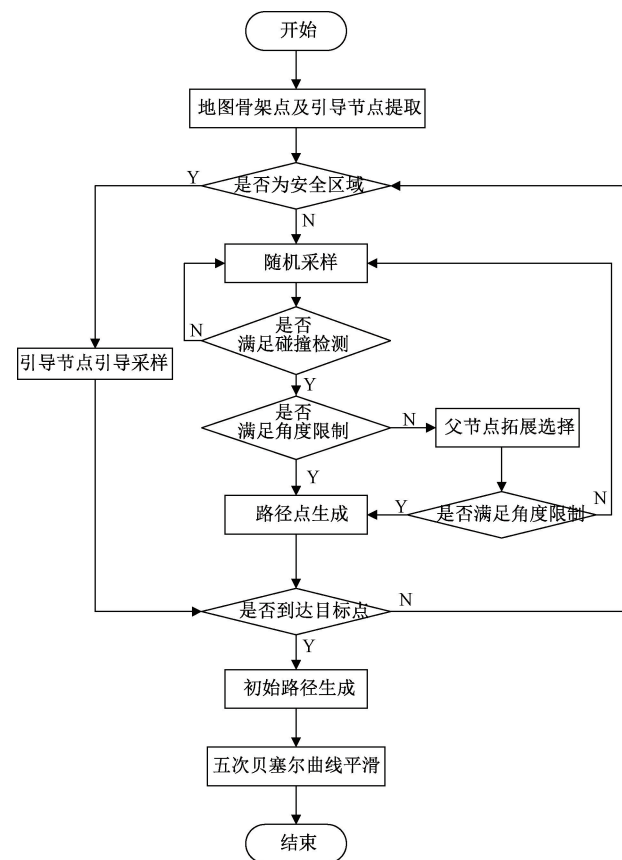


图 10 RSS_GN RRT 算法流程

Fig. 10 RSS_GN RRT algorithm flowchart

3 低速动态环境下的 RSS_GN RRT 算法

RSS_GN RRT 算法能够快速规划一条全局路径,而实际场景往往是动态且变化的,可能会存在先验地图中未标记的障碍物,例如临时的指路标志牌、行人等。这些障碍物可能会干扰基于栅格地图的路径规划,有时甚至引起规划路径与障碍物之间发生碰撞,正如图 11 所示。显然,单次全局路径规划在这种情况下是不够的。为了提高算法在动态场景中的适用性,本文将 RSS_GN RRT 算法扩展应用于局部路径规划中,以确保车辆在变化的环境中保持安全且高效地行驶。

本文利用全局路径信息、栅格地图信息,以及定位和感知模块所提供的实时数据,实现了局部避障功能。方案框架如图 12 所示,具体的局部避障流程图如图 13 所示。在车辆的行驶过程中,场景感知模块持续监测周围

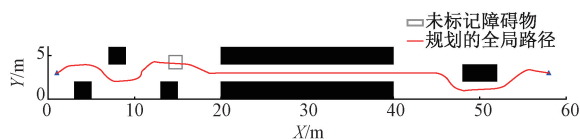


图 11 规划路径与未标记障碍物碰撞图

Fig. 11 Planned path and unmarked obstacle collision scenario diagram

环境的变化,将检测到的障碍物信息实时传送到路径规划模块。规划模块会将这些实时数据与栅格地图信息融合,调整局部栅格地图的状态,实现地图的动态重构。同时评估障碍物对车辆行进路径的潜在影响。如图 11 所示,若检测到障碍物对已规划的路径产生了干扰,则需要调整行驶路径。局部避障的具体过程如下:首先,算法会确定避障规划的起点和终点。为了预留一定的避障时间与空间,车辆会在遇到障碍物的位置开始调整路径。随后,再次采用 RSS_GN RRT 算法重新规划出一条绕开障碍物的局部路径。

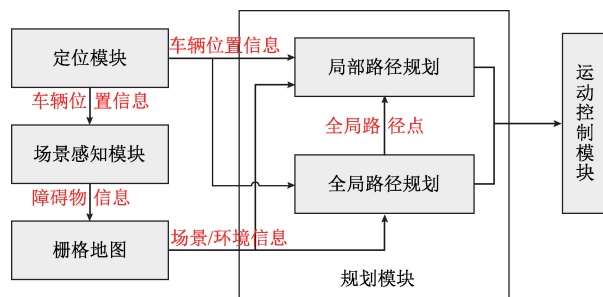


图 12 局部避障方案框架图

Fig. 12 Local obstacle avoidance scheme frame diagram

局部规划不仅可以有效地避开障碍物,还可保证车辆尽可能沿着原先规划的路径行驶,提高了算法在动态场景中的适应性和灵活性。然而,目前的避障方法将障碍物视为瞬时出现的静态对象,更适用于低速、变化缓慢的场景。在高速动态的场景中,由于环境的快速变化,路径规划模块需要实时、快速地调整路径,这可能导致路径剧烈变化,不利于车辆的运动控制。此外,在高速环境中,规划和控制模块之间的反应迟滞问题更加突出,这要求车辆具备更先进的轨迹预测和提前规划能力,解决这些问题也是未来的研究内容。

4 实验结果

4.1 仿真实验

为了验证 RSS_GN RRT 算法的性能,本文将传统 RRT、Goal_bias RRT^[11]、角度约束下的 RRT、Informed RRT^[9]、DR-RRT^[18]以及 RSS_GN RRT 算法在 3 种不同

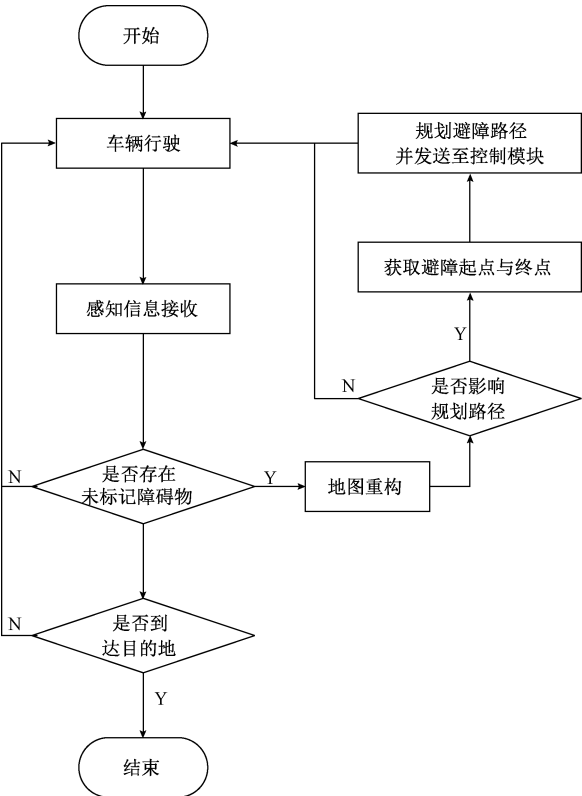


图 13 局部避障流程

Fig. 13 Local obstacle avoidance flowchart

的仿真环境下进行对比实验,仿真环境地图如图 14 所示。每种算法各进行 100 次重复实验,统计并分析不同算法的各种指标。

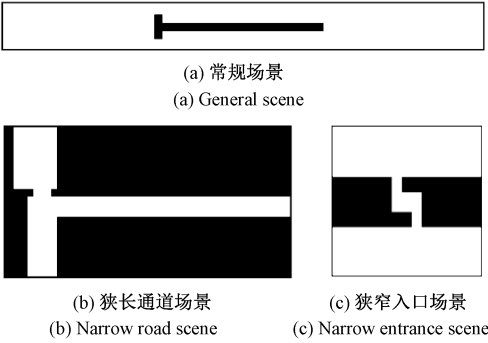


图 14 仿真实验地图

Fig. 14 Simulation experiment map

1) 常规场景中的仿真实验

图 15 展示了在常规场景下的路径规划结果,图 15 (a)、(b)、(c)、(d)、(e) 和 (f) 分别对应 RRT、Goal_bias RRT、角度约束下的 RRT、Informed RRT*、DR-RRT 和 RSS_GN RRT 算法的规划结果,表 1 展示了这些算法在不同性能指标上的对比数据。

如图 15(a) 所示,由于 RRT 算法盲目采样的特性,

其生成的路径曲折且复杂,并包含许多无用的节点。如图 15(b) 所示,Goal_bias RRT 算法利用目标导向策略减少了无用节点的产生,但路径质量较差。如图 15(c) 所示,角度约束下的 RRT 算法生成的路径更平滑,但采样角度的约束导致规划时间和迭代次数的大幅增加。图 15(d) 所示为 Informed RRT* 算法规划的结果,算法在初始路径生成后,通过在椭圆区域内进行迭代优化来提升路径质量,但这也增加了算法的时间消耗。图 15(e) 中,DR-RRT 算法通过沿地图骨架采样减少了对非目标区域的搜索,但采样点过度集中于骨架附近,生成的路径十分曲折。与此相比,图 15(f) 中的 RSS_GN RRT 算法表现出明显的性能优势。该算法利用引导节点导向和分区域采样的策略,能够迅速找到路径,减少了大量的无用节点和迭代次数。同时,改进的角度约束保证了规划路径的平滑性。

由表 1 可得,在常规场景中,RSS_GN RRT 算法相比于 RRT、Goal_bias RRT、角度约束下的 RRT、Informed RRT* 以及 DR-RRT 算法,规划时间分别减少了 54.3%、24.7%、71%、63.8% 和 55.4%,迭代次数分别减少了 84.9%、80.1%、95.5%、80.3% 和 89.5%,规划路径的平均曲率减小了 88.5%、82.1%、80%、70.8% 和 91%,且规划路径的平均长度最短。这些数据表明,RSS_GN RRT 算法在规划效率、迭代次数以及路径平滑性方面都有明显提升。

2) 狭长通道场景中的仿真实验

为了验证算法在狭长空间中的规划性能,本节设计了一种狭长的通道测试场景。图 16 为各算法的路径规划结果,表 2 展示了这些算法在不同性能指标上的对比数据。

如图 16(a)、(c)、(d) 所示,RRT、角度约束下的 RRT 以及 Informed RRT* 算法在规划过程中对许多非目标区域进行了搜索,这增加了迭代次数与耗时。如图 16(b) 所示,在目标点导向下,Goal_bias RRT 算法减少了对无关区域的搜索,提高了规划速度。但在起终点间有大面积不可通行区域的情况下,其目标点采样策略的效果受限,导致路径出现波动。图 16(e) 中的 DR-RRT 算法在较少的迭代次数和较短的规划时间内完成了路径规划,但生成的路径较为曲折。如图 16(f) 所示,RSS_GN RRT 算法在引导节点的引导下避免了对非目标区域的无效搜索,可快速生成路径。其平均规划时长仅为 0.025 s,平均迭代次数最少,仅 52 次,所生成路径的平均曲率仅为 0.06 m^{-1} ,路径质量最优。此外,在狭长通道场景中,角度约束下的 RRT 算法的规划成功率仅为 12%,而 RSS_GN RRT 算法为 100%。可见,RSS_GN RRT 算法通过父节点拓展选择的方法克服了角度约束导致规划成功率低下的问题。

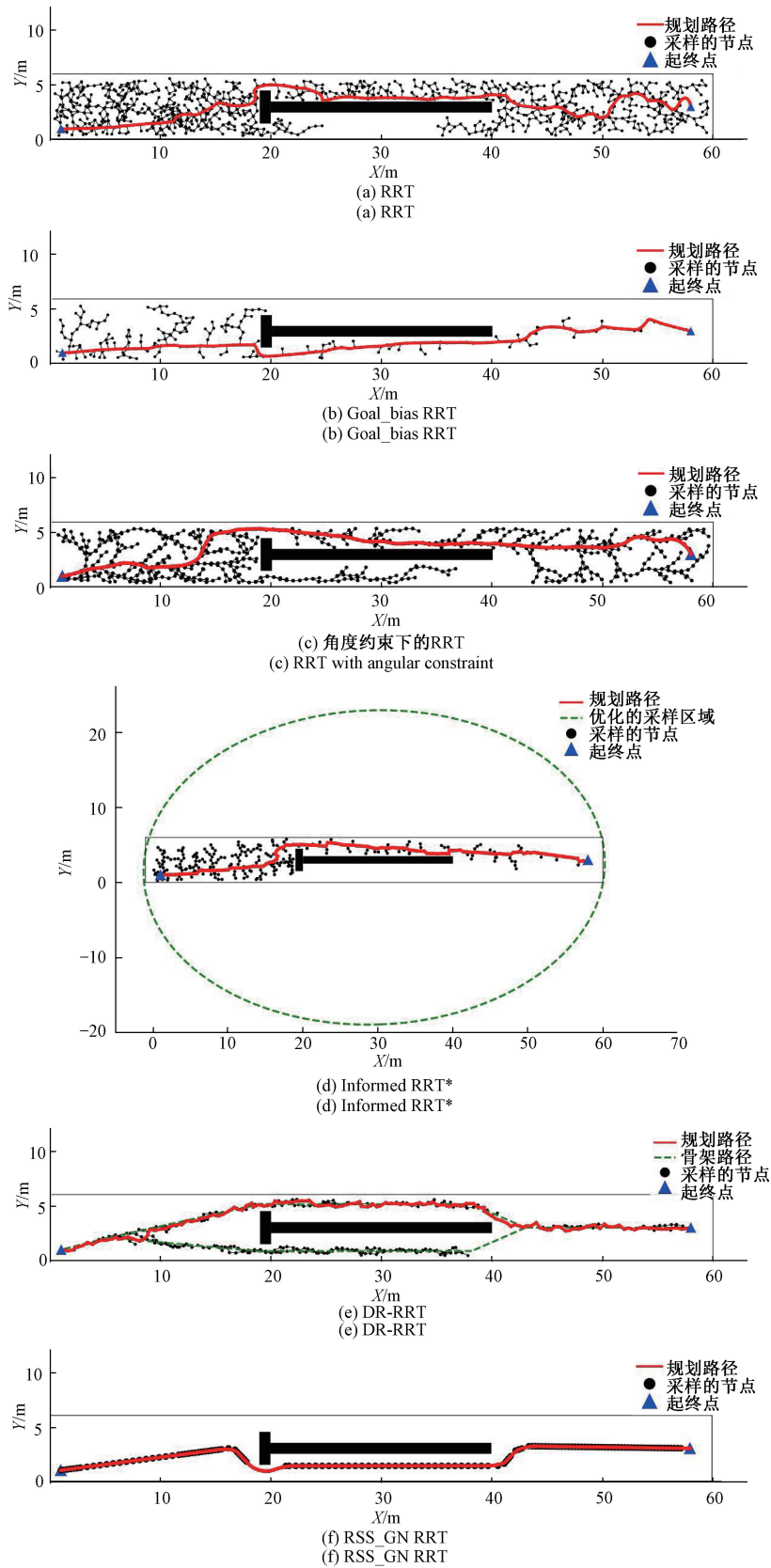


图 15 在常规场景中的仿真结果

Fig. 15 Simulation results in general scene

表 1 在常规场景中的实验结果对比

Table 1 Comparative experimental results in general scenarios

比较指标	RRT	Goal_bias RRT	角度约束下的 RRT	Informed RRT*	DR-RRT	RSS_GN RRT
规划成功率/%	100	100	24	100	100	100
平均规划时长/s	0.127	0.077	0.2	0.16	0.13	0.058
平均迭代次数	860	653	2915	660	1244	130
平均曲率/ m^{-1}	0.61	0.39	0.35	0.24	0.78	0.07
平均长度/m	63	60	59	60.34	59.84	58

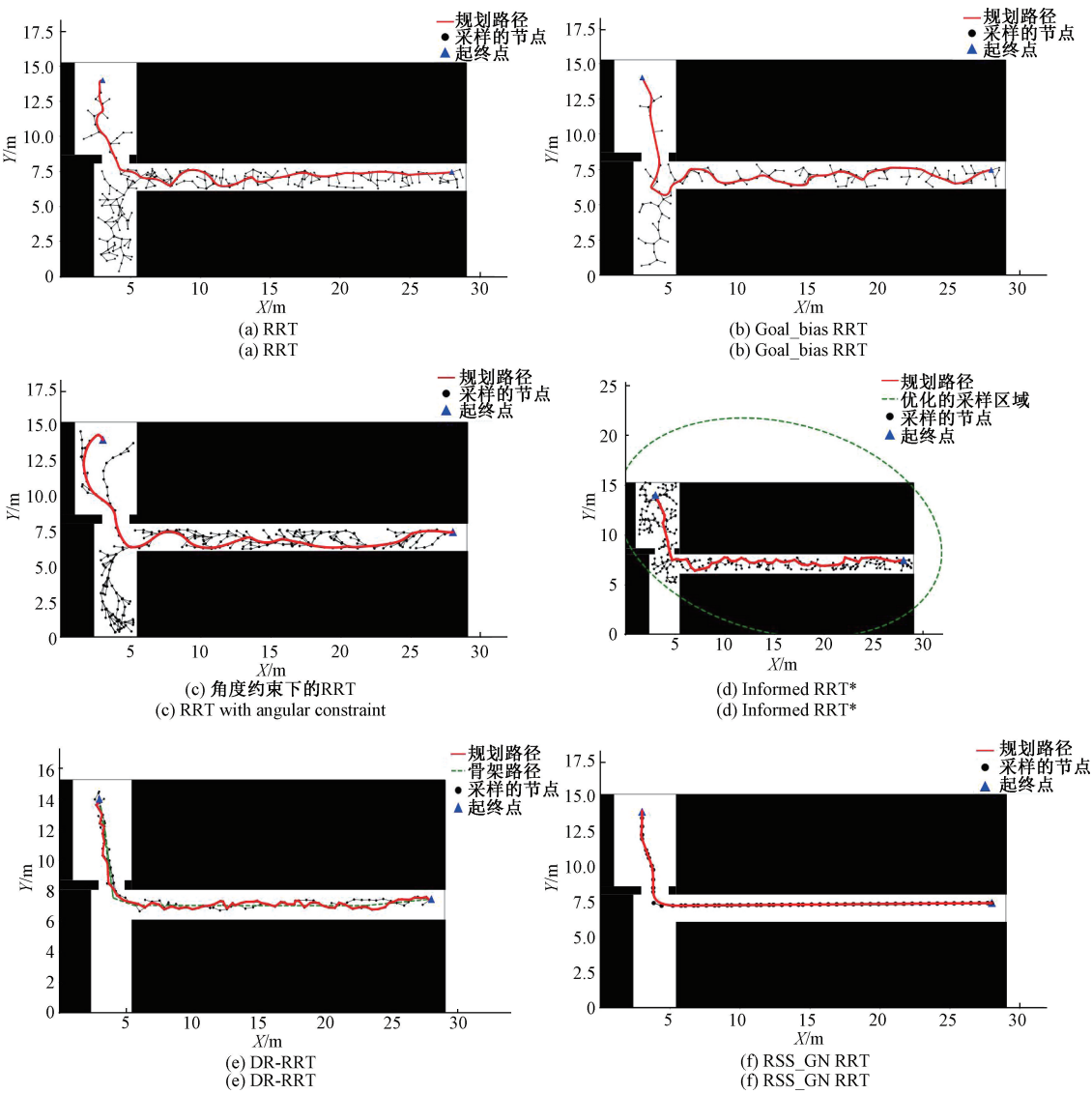


图 16 在狭长通道场景中的仿真结果

Fig. 16 Simulation results in the narrow road scene

由表 2 可得,在狭长的通道场景下,RSS_GN RRT 算法相比于 RRT、Goal_bias RRT、角度约束下的 RRT、Informed RRT * 以及 DR-RRT 算法,规划时间分别减少了 77.3%,51.9%,84.7%,98.8%和 60.3%,迭代次数分别减少了 95.9%,92%,98.3%,98.3%和 89.5%,路径的平

均曲率减小了 94.1%,93.2%,88.7%,91%和 92.9%,且规划路径的平均长度最短。结果验证了 RSS_GN RRT 算法在狭长空间中的适用性及高效性。

3) 狭窄入口场景中的仿真实验本节设置了一种狭窄入口的测试场景,自起点至终点的路径必须穿越一个狭

窄的通道,以检验算法寻找正确采样方向以及在受限通道中高效规划的能力。图 17 和表 3 分别展示了不同算法的规划结果图和性能指标数据。

表 2 在狭长通道场景中的实验结果对比

Table 2 Comparative experimental results in narrow road scene

比较指标	RRT	Goal_bias RRT	角度约束下的 RRT	Informed RRT*	DR-RRT	RSS_GN RRT
规划成功率/%	100	100	12	100	100	100
平均规划时长/s	0.11	0.052	0.163	2.088	0.063	0.025
平均迭代次数	1 271	651	3 004	3 000	493	52
平均曲率/ m^{-1}	1.02	0.88	0.53	0.67	0.84	0.06
平均长度/m	37	35	33	32.95	30.89	30

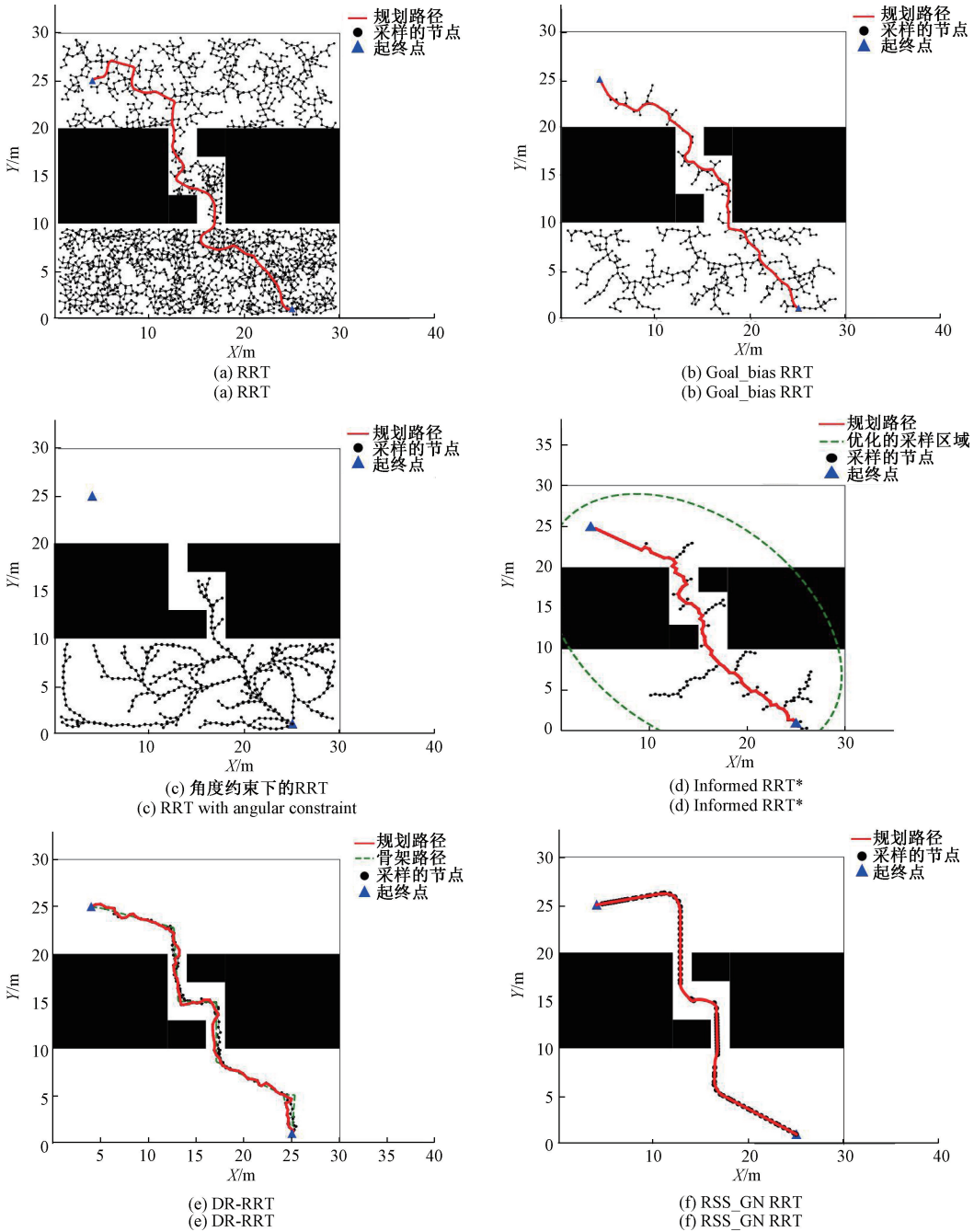


图 17 在狭窄入口场景下的仿真结果

Fig. 17 Simulation results in narrow entrance scene

如图 17(a)、(b)、(c)、(d) 所示, RRT、Goal_bias RRT、角度约束下的 RRT 以及 Informed RRT * 算法在寻找狭窄通道的入口时, 产生了大量的无用节点。这些算法的迭代次数多, 规划速度慢。在相同的迭代次数下, 角度约束下的 RRT 算法甚至未能完成规划任务。虽然, RSS_GN RRT 算法的成功率也未达 100%, 但相比于角度约束下的 RRT, 其成功率有了显著的提升。如图 17(e)、(f) 所示, DR-RRT 与 RSS_GN RRT 算法分别在地图骨架及引导节点的指引下, 可以迅速且准确地找到正确的采样方向并完成路径规划, 规划时长均小于 0.1 s, 大大提高了规划速度。此外, RSS_GN RRT 算法可根据区域状

态自适应地调整采样方式, 进一步加快了规划速度, 其规划耗时比 DR-RRT 算法更短。同时, 在采样角度约束的优化下, RSS_GN RRT 算法规划出的路径更为平滑, 展现出其在规划速度和路径质量上的优越性。

由表 3 可得, 在狭窄入口场景下, 相比于 RRT、Goal_bias RRT, Informed RRT * 以及 DR-RRT 算法, RSS_GN RRT 算法的规划时间依次减少了 86.5%, 76.5%, 96.4% 和 39.7%, 迭代次数分别减少了 93.3%, 92.1%, 91.9% 和 80.7%, 路径的平均曲率减小了 82%, 77.8%, 64.7% 和 80.6%。结果表明, RSS_GN RRT 算法能够在受限制的通道环境中完成高效且高质量的路径规划。

表 3 在狭窄入口场景下的实验结果对比

Table 3 Comparative experimental results in narrow entrance

比较指标	RRT	Goal_bias RRT	角度约束下的 RRT	Informed RRT *	DR-RRT	RSS_GN RRT
规划成功率/%	100	96	0	100	100	96
平均规划时长/s	0.327	0.187	—	1.207	0.073	0.044
平均迭代次数	1 727	1 462	—	1 427	595	115
平均曲率/m ⁻¹	1	0.81	—	0.51	0.93	0.18
平均长度/m	44	41	—	40.05	38.68	41

在 3 种不同的测试场景中, RSS_GN RRT 算法在规划时长、迭代次数及路径曲率方面都有显著提升。该算法不仅提高了规划效率, 还成功解决了角度约束带来的规划成功率下降的问题。此外, 在常规和狭长通道场景中, RSS_GN RRT 算法规划的路径长度最短。尽管在狭窄入口场景中, 路径长度不是最短的, 与其他算法相比仍差距不大, 但其在减少迭代次数、缩短规划时间和改善路径平滑性方面的性能更加优越。总体而言, RSS_GN RRT 算法在多种场景下均展现了其高效、准确和可靠的路径规划能力。

4.2 局部避障实测实验

为了验证算法在低速动态场景下的适用性, 本文对算法的局部避障能力进行了实测。测试采用了 HUNTER 阿克曼模型可编程的无人化地面车辆, 如图 18 所示, 其配备激光雷达、GNSS 接收机、视觉相机、电源、工控机等。工控机的环境信息如表 4 所示。

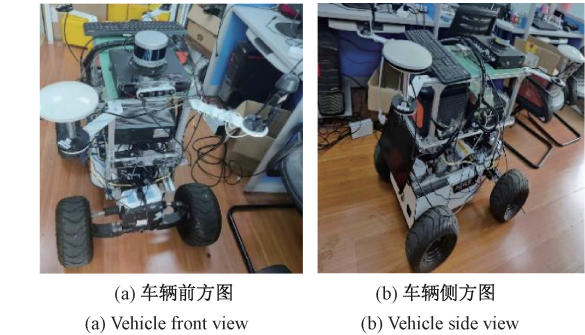


图 18 测试实验车辆图

Fig. 18 Test experimental vehicle diagram

表 4 工控机环境信息

Table 4 Environment information of industry-control-computer

名称	版本
Ubuntu	20.04
ROS	noetic
Python	3.7.16
Pytorch	1.13.1
CUDA	11.7
CPU	Intel(R) Core(TM) i9-10900
GPU	Nvidia Geforce RTX 3060

首先, 采用 RSS_GN RRT 算法完成全局路径规划, 车辆按照规划路径行驶, 全局路径规划结果如图 19 所示。

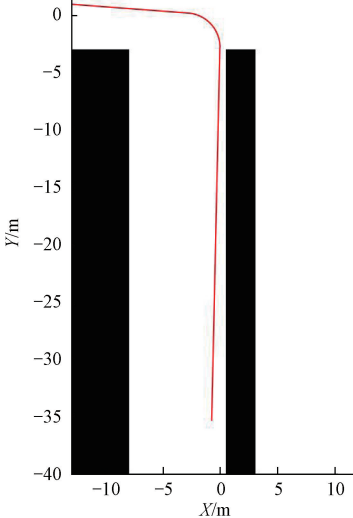


图 19 全局路径规划结果

Fig. 19 Global path planning results

在行进的过程中,随机设置目标障碍物,以测试算法的避障能力,实验结果如图 20 所示。由图 20 可知,场景中多次出现障碍物,规划模块可根据感知结果在地图中更新

障碍物信息,并生成避障路径。经测试,车辆均可成功地避开障碍物,并按照新规划的路径继续行驶。

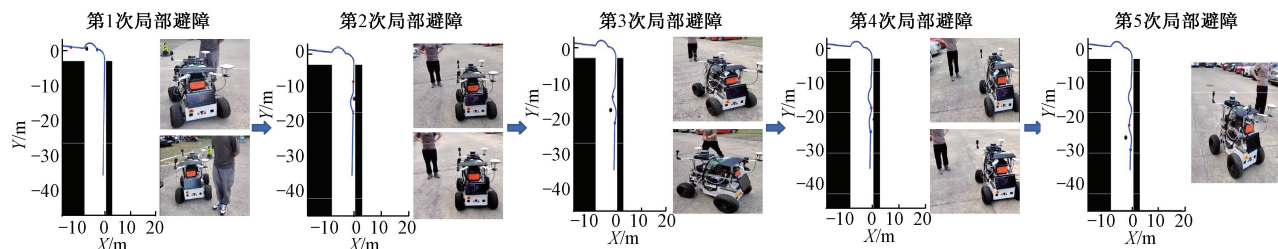


图 20 局部避障测试结果

Fig. 20 Local obstacle avoidance test results

5 结 论

针对 RRT 算法存在的收敛速度慢、规划路径质量差的问题,本文提出了一种改进的 RSS_GN RRT 算法。该算法利用栅格地图提取的引导节点为路径点采样提供准确的方向指引,从而降低算法的随机性,提高收敛速度。在路径点的采样过程中,算法根据区域状态自适应地调整采样方式,并引入采样角度约束,提高了路径质量,加快了算法在空闲区域的搜索速度。此外,采样角度约束导致算法的迭代次数显著上升,本文提出了一种父节点拓展选择的方法,有效地解决了这一问题。实验结果表明,在常规、狭窄通道和狭窄入口场景下,相比于其他算法,RSS_GN RRT 算法规划所需的时间最短,迭代次数最少,平均曲率最小,同时算法还能够有效地应对车辆行驶中遇到的障碍物。进一步提高复杂或极端场景下算法的规划成功率是未来的研究重点。

参考文献

- [1] 李永丹,马天力,陈超波,等. 无人驾驶车辆路径规划算法综述[J]. 国外电子测量技术, 2019, 38(6): 72-79.
LI Y D, MA T L, CHEN CH B, et al. Review of path planning algorithm for unmanned vehicles [J]. Foreign Electronic Measurement Technology, 2019, 38(6): 72-79.
- [2] 姜媛媛,张阳阳. 改进 8 邻域节点搜索策略 A* 算法的路径规划[J]. 电子测量与仪器学报, 2022, 36(5): 234-241.
JIANG Y Y, ZHANG Y Y. Improved path planning of A* algorithm of domain node search strategy 8 [J]. Journal of Electronic Measurement and Instrumentation, 2022, 36(5): 234-241.
- [3] 侯宇翔,高焕兵,汪子健,等. 改进 RRT 的移动机器人

路径规划算法[J]. 电子测量技术, 2022, 45(16): 47-52.

HOU Y X, GAO H B, WANG Z J, et al. Improved RRT for mobile robot path planning algorithm [J]. Electronic Measurement Technology, 2022, 45(16): 47-52.

- [4] 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.

- [5] 魏彤,龙琛. 基于改进遗传算法的移动机器人路径规划[J]. 北京航空航天大学学报, 2020, 46(4): 703-711.

WEI T, LONG CH. Path planning for mobile robot based on improved genetic algorithm [J]. Journal of Beijing University of Aeronautics and Astronautics, 2020, 46(4): 703-711.

- [6] 张亮,王鑫,张建锋. 多障碍场景下改进 RRT 算法路径规划[J]. 计算机工程与设计, 2023, 44(6): 1706-1713.

ZHANG L, WANG X, ZHANG J F. Improved RRT algorithm path planning in multi-obstacle scenarios [J]. Computer Engineering and Design, 2023, 44(6): 1706-1713.

- [7] 杜传胜,高焕兵,侯宇翔,等. 同根双向扩展的贪心 RRT 路径规划算法[J]. 计算机工程与应用, 2023, 59(21): 312-318.

DU CH SH, GAO H B, HOU Y X, et al. Greedy RRT path planning algorithm with same root bidirectional extension [J]. Computer Engineering and Applications, 2023, 59(21): 312-318.

- [8] 林依凡,陈彦杰,何炳蔚,等. 无碰撞检测 RRT* 的移动机器人运动规划方法[J]. 仪器仪表学报, 2020, 41(10): 257-267.
LIN Y F, CHEN Y J, HE B W, et al. Non-collision checking RRT* algorithm for mobile robot motion planning[J]. Chinese Journal of Scientific Instrument, 2020, 41(10): 257-267.
- [9] GAMMEEL J D, SRINVASA 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, Chicago, 2014:2997-3004.
- [10] NASIR J, ISLAM F, MALIK U, et al. RRT*-SMART: A rapid convergence implementation of RRT* [J]. International Journal of Advanced Robotic Systems, 2013, 10(7): 1-12.
- [11] LAVALLE S M, KUFFNER JR J J. Randomized kinodynamic planning[J]. The International Journal of Robotics Research, 2001, 20(5): 378-400.
- [12] 刘奥博,袁杰. 目标偏置双向 RRT* 算法的机器人路径规划[J]. 计算机工程与应用, 2022, 58(6): 234-240.
LIU AO B, YUAN J. Robot path planning based on goal biased bidirectional RRT* algorithm [J]. Computer Engineering and Applications, 2022, 58(6): 234-240.
- [13] 阮晓钢,周静,张晶晶,等. 基于子目标搜索的机器人目标导向 RRT 路径规划算法[J]. 控制与决策, 2020, 35(10): 2543-2548.
RUAN X G, ZHOU J, ZHANG J J, et al. Robot goal guide RRT path planning based on sub-target search[J]. Control and Decision, 2020, 35(10): 2543-2548.
- [14] CHEN X, FAN J. UAV trajectory planning based on APF-RRT* algorithm with goal-biased strategy[C]. 2022 34th Chinese Control and Decision Conference (CCDC). IEEE, 2022: 3253-3258.
- [15] 李伟东,李乐. 基于改进 RRT 算法的无人车路径规划[J]. 计算机测量与控制, 2023, 31(1): 160-166.
LI W D, LI L. Path planning of unmanned vehicle based on improved RRT algorithm[J]. Computer Measurement & Control, 2023, 31(1): 160-166.
- [16] 赵超力,马行,张春涛,等. 基于引力场引导的 RRT-connect 路径规划算法[J]. 电子测量技术, 2021, 44(22): 44-49.
ZHAO CH L, MA X, ZHANG CH T, et al. RRT-connect path planning algorithm based on gravitational field guidance[J]. Electronic Measurement Technology, 2021, 44(22): 44-49.
- [17] 司徒华杰,雷海波,庄春刚. 动态环境下基于人工势场引导的 RRT 路径规划算法[J]. 计算机应用研究, 2021, 38(3): 714-717,724.
SITU H J, LEI H B, ZHUANG CH G. Artificial potential field based RRT algorithm for path planning in dynamic environment [J]. Application Research of Computers, 2021, 38(3): 714-717,724.
- [18] DENNY J, SANDSTRÖM R, BREGGER A, et al. Dynamic region-biased rapidly-exploring random trees [C]. Algorithmic Foundations of Robotics XII: Proceedings of the Twelfth Workshop on the Algorithmic Foundations of Robotics. Springer International Publishing, 2020: 640-655.
- [19] 董璐,熊爱玲. 基于改进 RRT*-Smart 的复杂动态环境下的无人艇路径规划[J]. 智能科学与技术学报, 2022, 4(2): 264-276.
DONG L, XIONG AI L. Path planning for unmanned surface vehicle in complex dynamic environment based on improved RRT*-Smart [J]. Chinese Journal of Intelligent Science and Technology, 2022, 4(2): 264-276.
- [20] 赵港. 改进 RRT* 算法的智能车辆路径规划[J]. 汽车实用技术, 2021, 46(22): 41-43.
ZHAO G. Vehicle path planning based on improved RRT* algorithm [J]. Automobile Applied Technology, 2021, 46(22): 41-43.
- [21] 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.
- [22] 甘新基. 基于 Bézier 曲线的差速驱动机器人混合避障路径规划算法[J]. 吉林大学学报(理学版), 2021, 59(4): 943-949.
GAN X J. Path planning algorithm for hybrid obstacle avoidance of differential drive robot based on Bezier curve[J]. Journal of Jilin University (Science Edition), 2021, 59(4): 943-949.
- [23] 李军,周伟,唐爽. 基于自适应拟合的智能车换道避障轨迹规划[J]. 汽车工程, 2023, 45(7): 1174-1183,1199.
LI J, ZHOU W, TANG SH. Lane change and obstacle avoidance trajectory planning of Intelligent vehicle based on adaptive fitting[J]. Automotive Engineering, 2023,

45(7): 1174-1183, 1199.

作者简介



she is a M. Sc. candidate of Southeast University. Her main research interest includes unmanned vehicle path planning.

王萍, 2022 年于中国计量大学获得学士学位, 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: 987757833@qq.com

Wang Ping received her B. Sc. degree from China Jiliang University in 2022. Now



潘树国 (通信作者), 2007 年于东南大学获得博士学位, 现为东南大学教授, 博士生导师, 主要研究方向为 GNSS 高精度定位与环境智能感知。

E-mail: psg@seu.edu.cn

Pan Shuguo (Corresponding author) received his Ph. D. degree from Southeast University in 2007. Now he is a professor, Ph. D. supervisor in Southeast University. His main research interests include GNSS high-precision positioning and environmental intelligent sensing.