

基于 IRRT^{*}-Connect 的自适应路径规划算法^{*}

马晓群 王 昊 刘 磊 李 树

(辽宁工业大学 锦州 121001)

摘 要: 针对在复杂障碍环境下 IRRT^{*}-Connect 算法采样目的性弱、收敛速度慢和路径优化效果差的问题,本文提出了一种依赖环境复杂度的基于 IRRT^{*}-Connect 的自适应路径规划算法。该算法采用 IRRT^{*}-Connect 算法进行初始路径规划以提高初次路径规划效率;其次,该算法引入采样约束概率 p 对可采样区域进行限制,增强算法采样目的性;最后设计基于环境障碍系数的步长计算方法以实现扩展步长自适应动态调整,增强算法通过复杂环境的能力。通过多组实验对比表明,在复杂环境下,算法节点数减少了 9.75%,且路径长度减少了 20.82%,规划时间缩短了 3.08%,证明本文所改进的 IRRT^{*}-Connect 自适应步长路径规划算法具有较强的适应环境能力,节点利用率高,规划效果更佳。

关键词: 路径规划;Informed-RRT^{*};自适应步长;采样约束

中图分类号: TP242;TN081 **文献标识码:** A **国家标准学科分类代码:** 520.60

Adaptive path planning algorithm based on IRRT^{*}-Connect

Ma Xiaqun Wang Hao Liu Lei Li Shu

(Liaoning University of Technology, Jinzhou 121001, China)

Abstract: In response to the challenges of slow convergence and suboptimal path optimization in complex obstacle environments, this paper presents an adaptive path planning algorithm based on IRRT^{*}-Connect that is tailored to the environmental complexity. This algorithm combines Informed-RRT^{*} and RRT-Connect, namely IRRT^{*}-Connect algorithm for initial path planning to improve the efficiency of initial path planning; additionally, it introduces a sampling constraint probability p to confine sampled areas and augment the purposefulness of sampling. Furthermore, a step length calculation method based on environmental obstacle coefficients is devised to dynamically adjust extension step lengths, thereby bolstering the algorithm's adaptability in traversing complex environments. Through the comparison of multiple sets of experiments, we show that the number of algorithm nodes is reduced by 9.75%, and the path length is reduced by 20.82%, and the planning time is shortened by 3.08%, which proves that the improved IRRT^{*}-Connect adaptive step path planning algorithm has a strong ability to adapt to the environment, high node utilization, and the planning effect is better.

Keywords: path planning; informed-RRT^{*}; self-adaptive step size; sampling constraint

0 引 言

随着第四次工业革命的到来,各领域中智能化机器人大量投入使用,而路径规划作为机器人学的基础问题之一^[1],更为广泛的应用于机械臂轨迹规划^[2]、工业巡检^[3]、自动驾驶^[4]等场景之中。路径规划是自主移动机器人的一个重要组成部分,它的任务就是在具有障碍物的环境内按照一定的评价标准,寻找一条从起始状态(包括位置和姿

态)到达目标状态(包括位置和姿态)的无碰撞路径^[5]。当前在已存路径规划算法中,不受环境类型限制的采样型算法已广泛应用于各种复杂环境下的路径规划问题,具有较高的鲁棒性和可靠性。作为采样型路径规划算法中的典型代表快速探索随机树(RRT-rapidly-exploring random trees, RRT)算法^[6]于1998年由美国爱荷华州立大学的 La valle 教授首次提出。RRT 算法以一个初始点作为根节点,通过对未探测区域进行随机采样,进而增加叶节点的方式

收稿日期:2024-05-20

*基金项目:国家自然科学基金重点项目(62333009)、辽宁省“揭榜挂帅”技术攻关项目(2023JH1/10400092)、辽宁省教育厅面上项目(JYTM20230837)资助

生成随机扩展树,直至存在叶节点到达目标点为止,由此即可获得由树节点构成的连接起止点的路径信息。因其随机性强,规划效率快的优点,RRT 算法在近年得到了广泛应用与发展。

针对 RRT 算法收敛速度较慢,搜索效率较低的问题,Kuffner 等^[7]提出了一种简单有效的随机化算法,即 RRT-connect 算法,该方法的工作原理是建立两个快速探索随机树植根于开始位置和目标位置,每棵树依次轮流探索周围空间,并通过使用简单的贪婪启发式算法相互靠近加快搜索效率;Karaman 等^[8]提出快速探索随机数改进算法(rapidly-exploring random trees star, RRT*),在 RRT 节点扩展基础上添加随机几何图与剪枝优化理论,确保随机树的节点都能收敛到当前最优值。但当探测区域较大,环境较为复杂时,RRT* 算法会产生数量庞大的节点信息,导致计算量以及内存消耗均大幅增加,使得算法规划速率较差。因此提高 RRT* 算法的有效节点数量,以减少算法计算量,降低内存消耗,是该算法优化研究的主要方向。Nasir 等^[9]提出了基于智能采样的 RRT*-smart 算法以提高收敛速度和降低计算量,但因其牺牲了随机探索特性,导致更优解丢失,以至于在不同的环境下应用时偏执率需人为进行调整,该算法自适应性较差。在此基础之上,Mashayekhi 等^[10]提出了混合 RRT 算法,通过双树搜索找到初始解,进而将起始树和目标树合二为一,实现单向搜索树的信息采样,以优化当前解。为了获得更优的初始解和更快的收敛速度,Liao 等^[11]提出了以随机点创建父节点的改进型 RRT* 算法即 F-RRT* 算法。基于 F-RRT* 算法,杨敏豪等^[12]采用双向搜索方式,加速 F-RRT* 算法收敛速度,同时消除扩展过程中的冗余点,引入启发函数,对连接点进行优化,提高了路径质量。

与此同时,不断有学者对 RRT 算法各步骤进行优化,加速该算法搜索效率。范伟伦等^[13]提出了一种改进 RRT 算法,该算法通过步长的动态确定,解决了生长树在靠近障碍区域时所存在局限性的问题,通过在节点生长时增加自适应目标引力,将节点的生长方向偏引至目标方向,降低了 RRT 算法随机性。陈都等^[14]采用融合采样点概率及加入目标点引力的方式进行树节点扩张,并引入动态步长策略提高了搜索效率。顾子倡等^[15]在随机树的待扩展节点选取上引入目标启发信息和对动态扩展随机树进行剪枝和重构方式,达到加速算法收敛和快速避开威胁、生成安全航路的目的。施成龙等^[16]为缩短 RRT 算法规划时间,在原始基础上加入了动态步长策略,根据周围环境中障碍物分布情况设置步长,同时结合 Dijkstra 算法对路径进行规划,使规划效率有较大提升。上述文献均只考虑提高算法搜索效率问题,并未研究最优路径规划问题。

最优路径规划^[17]问题指对最优路径的搜索,RRT 路径规划算法在规划可行路径时,因其概率完备性该算法总能找到一条可行路径,但同时由于概率随机,所规划的路径

σ 并不一定是最优路径 σ^* 。

为使规划结果最优,Gammell 等^[18]提出了 Informed-RRT* 算法,该算法采样点位于 RRT* 所生成路径附近的椭圆区域,在迭代过程中随着路径代价的缩小,采样区域逐渐缩小,以更高效率获取最优路径。Mashayekhi 等^[19]基于该算法提出一个单查询双向规划方法的最佳运动规划问题,称为 Informed-RRT*-Connect,该算法结合 Informed-RRT* 与 RRT*-Connect 算法,在初次规划时采用双向搜索加速搜索效率,通过限制采样状态空间使得所获路径渐近最优。然而,限制状态空间为有关于所获得的最短路径的函数,在找到解之前,算法无法将状态空间限制为子集,算法需要更多时间来找到初始解的缺点,此外,其采样过程由于其随机性较强,采样目的性较弱,无法以较快时间到达目标点。当前已有算法研究中,基于该算法的改进方法不可多得。故其初次规划算法仍具有较大改进空间。

因此本文结合 Informed-RRT* 算法与 RRT-Connect 算法作为基础规划算法,同时提出一种采样区域约束方案,以增强节点利用率;此外,为提高算法的地图适应能力,本文提出一种自适应步长调整策略,算法能够根据环境障碍复杂程度的不同,动态调节步长,使得算法狭窄区域规划能力增强,空旷区域规划速率加快,路径代价更优。

1 基础知识

1.1 RRT-Connect 改进算法介绍

针对传统 RRT 算法通过狭窄、复杂地区时容易陷入局部解,难以规划出最优路径的问题,研究者在 RRT-Connect 引入 Connect 启发式算法,将传统的扩张函数替换为一种贪婪策略,从而在探索可行路径时具有更高的效率,在找到可行路径时迅速连接起始状态和目标状态,使得 RRT-Connect 能够更快地收敛到可行路径。

RRT-Connect 算法中两颗随机树交替生长,当算法进入本轮迭代时,首先通过随机采样函数 $Sample()$ 获取随机点 x_{rand} ,使用扩展函数 $Extend()$ 拓展 A 树新节点 x_{new} ,在 B 树中寻找离 x_{new} 最近的节点 $Close Node$,通过 $Connect()$ 函数判断二者是否可以满足连通条件,若无法连接则对 B 树进行下一次迭代扩展,直至两树连通。算法流程表如表 1 所示。

1.2 椭圆子集采样算法介绍

Informed-RRT* 算法对其 2 次采样区域进行限制,确保采样点均在最优路径附近的椭圆区域,即 $x_{ellipse} \in X_{ellipse}$,在获取代价更小路径后修改椭圆区域参数,使得在迭代过程中随着路径代价的缩小,采样区域随之缩小,获取最优路径。

由标准椭圆方程类推,在 Informed RRT* 算法中,以起点 x_{start} 、终点 x_{goal} 作为椭圆焦点,长轴长度 a 取初始路径长度 c_{best} 的一半,即 $a = c_{best}/2$,取 $c = c_{min}/2$,则短轴长度 $b = \sqrt{c_{best}^2 - c_{min}^2}/2$,至此椭圆方程所有参数获取完毕。

表 1 RRT-Connect 算法流程

Table 1 The algorithm process of RRT Connect

Algorithm 1 RRT-Connect algorithm

```

1:  $V_a \leftarrow \{x_{start}\}; E_a \leftarrow \emptyset;$ 
2:  $V_b \leftarrow \{x_{goal}\}; E_b \leftarrow \emptyset;$ 
3:  $G_a \leftarrow (V_a, E_a); G_b \leftarrow (V_b, E_b);$ 
4: for  $i=1$  to  $n$  do
5:    $x_{rand} \leftarrow Sample();$ 
6:   if  $Extend(G_a, x_{rand}) \neq Trapped$  then
7:     if  $Connect(G_b, x_{new}) = Reached$  then
8:       return  $G_a, G_b;$ 
9:     end if
10:  end if
11:   $Swap(G_a, G_b);$ 
12: end for
13: return  $G_a, G_b;$ 

```

Informed RRT* 算法采样区域如图 1 所示。

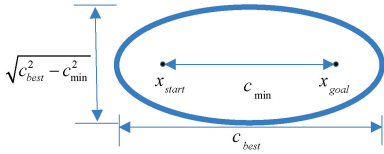


图 1 椭圆采样区域

Fig. 1 Elliptical sampling area

从 n 维的单位球中均匀分布样本转换到超椭圆中均匀分布的样本所需的变换矩阵 L 如式(1)所示。

$$L = \text{diag}\left\{\frac{c_{best}}{2}, \frac{\sqrt{c_{best}^2 - c_{min}^2}}{2}, \dots, \frac{\sqrt{c_{best}^2 - c_{min}^2}}{2}\right\}; \quad (1)$$

从椭圆坐标系到世界坐标系的旋转矩阵 C 如式(2)所示。

$$C = U \text{diag}\{1, \dots, 1, \det(U)\det(V)\}V^T \quad (2)$$

其中, $\det(U), \det(V)$ 分别为矩阵 U, V 的行列式, U, V 由矩阵 M 通过奇异值分解得到的酉矩阵, 如式(3)所示。

$$U \Sigma V^T \equiv M \quad (3)$$

其中, M 为世界坐标系下超椭圆体横轴 a_1 与单位矩阵 I_1 第一列的外积, 如式(4)所示。

$$M = a_1 \times I_1 \quad (4)$$

通过运算, 最终得到世界坐标系下采样域样本 $x_f \sim u(X_f)$ 如式(5)所示。

$$x_f = CLx_{ball} + x_{center} \quad (5)$$

2 算法改进

2.1 IRRT*-Connect 算法

为平衡 x_{star} 与 x_{goal} 两点周围节点数量, 提高搜索效率, 采用 IRRT*-Connect 作为搜索算法完成规划任务。IRRT*-Connect 是在 Informed-RRT* 算法的基础上引入 RRT-Connect 双向搜索算法。首先通过 $Sample()$ 采样函

数获取随机点 x_{rand} , 通过 $Nearest$ 函数获取搜索树 G_a 中距离 x_{rand} 最近的节点 $x_{nearest}$, $Steer()$ 函数根据两节点信息以及步长确定出新节点 x_{new} 位置, 将 x_{new} 加入到搜索树 G_a 中, 查找搜索树 G_a 中距 x_{new} 距离小于一定半径的所有邻近节点集合 X_{near} , 以此计算各个邻近点 x_{near} 到 x_{start} 的路径代价与 x_{new} 到 x_{near} 的路径代价, 对父节点进行修改。为进一步减小路径代价, 为随机树 G_a 进行重布线。在完成上述步骤后, 检测 G_b 中距离 x_{new} 最近的节点是否可以与 G_a 直接建立连接, 若可以则本次路径规划完毕, 将所有可行路径中路径代价最小的代价值作为 c_{best} , 以 G_b 为搜索树进入下一次迭代, 循环交换搜索树, 直至达到最大迭代次数。IRRT*-Connect 算法的流程表如表 2 所示。

表 2 IRRT*-Connect 算法流程

Table 2 The algorithm process of IRRT*-Connect

Algorithm 2 Informed RRT*-Connect

```

1:  $V_a \leftarrow \{x_{start}\}; E_a \leftarrow \emptyset;$ 
2:  $V_b \leftarrow \{x_{goal}\}; E_b \leftarrow \emptyset;$ 
3:  $G_a \leftarrow (V_a, E_a); G_b \leftarrow (V_b, E_b);$ 
4: for  $i=1$  to  $n$  do
5:    $c_{best} \leftarrow \min_{x_{soln} \in X_{soln}} \{Cost(x_{soln})\};$ 
6:    $x_{rand} \leftarrow Sample(c_{best}, c_{min}, x_{start}, x_{goal});$ 
7:    $x_{nearest} \leftarrow Nearest(G_a, x_{rand});$ 
8:    $x_{new} \leftarrow Self-Adaption-Steer();$ 
9:   if  $CheckPath((x_{nearest}, x_{new}))$  then
10:      $G_a \leftarrow G_a \cup \{x_{new}\};$ 
11:      $X_{near} \leftarrow Near(G_a, x_{new}, r_{RRT*});$ 
12:      $x_{min} \leftarrow x_{nearest};$ 
13:      $c_{min} \leftarrow Cost(x_{min}) + DistCost(x_{nearest}, x_{new});$ 
14:     for  $\forall x_{near} \in X_{near}$  do
15:        $c_{new} \leftarrow Cost(x_{new}) + DistCost(x_{near}, x_{new});$ 
16:        $x_{min} \leftarrow x_{near};$ 
17:        $c_{min} \leftarrow c_{new};$ 
18:     end for
19:      $c_{min} \leftarrow Cost(x_{min}) + DistCost(x_{nearest}, x_{new});$ 
20:     for  $\forall x_{near} \in X_{near}$  do
21:        $c_{near} \leftarrow Cost(x_{near});$ 
22:        $c_{new} \leftarrow Cost(x_{new}) + DistCost(x_{near}, x_{new});$ 
23:     end for
24:     end if
25:     if  $Connect(G_b, x_{new}) = Reached$  then
26:       return  $G_a, G_b;$ 
27:     end if
28:      $Swap(G_a, G_b);$ 
29:   end for
30:   return  $G_a, G_b;$ 

```

其中, $Cost()$ 函数可追溯指定可行路径的总代价; $Nearest()$ 函数通过依次计算已存在搜索树中的节点到当前产生的随机节点 x_{rand} 的距离代价, 获取距离节点 x_{rand} 距离代价最小的节点, 记为 $x_{nearest}$; $Check_path()$ 函数可检测所给的两点位置间是否存在障碍, 即是否可以建立连接; $Near()$ 函数可寻找临近节点 x_{near} , 将其存储在临近节点集合 X_{near} 中; $Dist_Cost()$ 函数可计算输入的两点间欧氏距离, 即路径代价; $Swap()$ 函数更换搜索树, 使得两颗搜索树交替生长; $Connect()$ 函数引入贪婪算法, 检测当前节点 x_{new} 是否可以与搜索树 G_b 中部分节点直接建立连接。

$Sample()$ 函数为改进采样函数, 在 Informed RRT* 椭圆区域采样的基础上进一步对采样区域进行有效约束; $Adapt_Steer()$ 函数根据随机节点 x_{rand} 、最近节点 $x_{nearest}$ 的信息以及步长信息获取新增长节点 x_{new} 位置信息, 此处步长采用一种基于采样点 x_{rand} 与最近节点 $x_{nearest}$ 区间障碍程度的自适应步长算法; 均于后文进行介绍。

2.2 采样区域优化方案

为解决 Informed-RRT* 算法所采用的椭圆子集二次采样方法在采样时, 出现多数采样点背离目标采样, 节点利用率低, 规划效率差的问题, 本文提出一种对椭圆子集进一步进行限制的采样方案, 在初始路径规划完成后, 引入约束概率 p , 令算法按概率 p 在二次约束区域内采样, 该采样约束区域示意图如图2所示, 使用采样区域约束方案的 $Sample()$ 函数算法流程如表3所示。



图2 采样区域示意图

Fig. 2 Schematic diagram of sampling area

表3 Sample 函数算法流程

Table 3 Algorithm flow of Sample function

Algorithm 3 Sample Function	
1:	function $Sample()$
2:	if $c_max < Inf$
3:	$p \leftarrow probability()$;
4:	if $rand < p$
5:	$Rand_node \leftarrow Min_Eil_sample()$
6:	else
7:	$Rand_node \leftarrow Elliptic_sample()$;
8:	end if
9:	else
10:	$Rand_node \leftarrow Rand_sample()$;
14:	end if
15:	end function

其中, $probability()$ 函数功能为根据当前最优路径代价值与起止点间欧氏距离计算约束概率 p , 公式为

$$p = (c_min / c_best) \times \zeta \quad (6)$$

其中, ζ 为约束系数, 采样点落在二次约束区域内的概率与 ζ 值成正比。 $Min_Eil_sample()$ 函数为在缩小后椭圆区域子集内进行采样的具体采样函数; $Elliptic_sample()$ 函数为在椭圆区域子集内进行采样的具体采样函数; $Rand_sample()$ 函数在全区域内进行采样的具体采样函数。

2.3 自适应步长策略算法

针对 Informed-RRT* 系列算法在搜索过程中若给定步长 $step$ 过大, 当面临复杂障碍区域规划任务时难以获取最优路径; 若给定步长 $step$ 过小, 而搜索区域障碍较为简单时虽然可以获取最优路径, 但所需采样次数过多, 搜索效率过低的问题, 本文采用一种基于采样点 x_{rand} 与最近节点 $x_{nearest}$ 区间障碍程度的自适应步长策略算法, 根据获取采样点 x_{rand} 与最近节点 $x_{nearest}$ 间的障碍复杂程度初步确定步长 $step$, 根据采样点 x_{rand} 大幅偏离目标点 x_{goal} 的情况对步长进行修正, 以高效获得可行采样点。自适应步长扩展如图3所示。

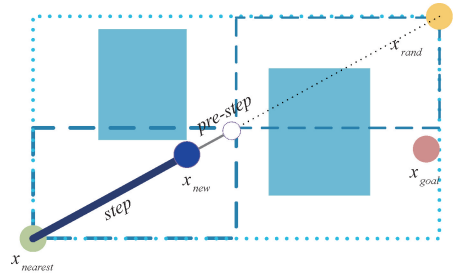


图3 自适应步长扩展图

Fig. 3 Diagram of adaptive step size extension

自适应步长调节策略具体方法如下: 在获取采样点 x_{rand} 与最近节点 $x_{nearest}$ 信息后以采样点 x_{rand} 与最近节点 $x_{nearest}$ 的中点为顶点, 划分出障碍主要区域 $Local_Map$, 以 $x_{nearest}$ 附近区域障碍密集程度为第一判断标准, 计算障碍系数 μ 。同时为避免产生采样点 x_{rand} 大幅偏离目标点 x_{goal} 的情况, 该算法在获取初步步长后引入修正系数 λ , 此系数根据最近节点 $x_{nearest}$ 分别与采样点 x_{rand} 与目标点 x_{goal} 间欧式距离不同可调, 若距离目标点较远则根据修正系数 λ 适当增加步长, 令修正系数 λ 的值为距目标点距离代价与距采样点距离代价的比值成正比例关系, 最终步长选取修正步长、初始步长以及最近节点 $x_{nearest}$ 与采样点 x_{rand} 间欧氏距离 3 者中最小值, 以进一步确保采样点靠近目标点。引入自适应步长策略的 $Adapt_Steer()$ 函数算法流程如表4所示。

其中, $Divide_Map()$ 函数功能为将以采样点 x_{rand} 与最近节点 $x_{nearest}$ 作为对角线的部分矩形地图 $Local_Map$ 从整体地图中提取出来; $Obs_area()$ 函数功能为计算区域内障碍所占面积; $Size()$ 函数功能为计算 $Local_Map$ 区域总面

表 4 Adapt Steer 函数算法流程

Table 4 Algorithm flow of Adapt Steer function

Algorithm 4 Adapt Steer Function	
1:	function Adapt Steer (i)
2:	$cost_rand \leftarrow Dist\ Cost(x_{nearest}, x_{rand})$;
3:	$cost_goal \leftarrow Dist\ Cost(x_{nearest}, x_{goal})$;
4:	$Local\ Map \leftarrow Divide\ Map()$;
5:	$\mu \leftarrow Obs_area(Local\ Map)$;
6:	$Pre\ step \leftarrow Adaption\ step(\mu)$;
7:	$\lambda = cost_goal / cost_rand$;
8:	$step = \min(pre\ step \times \lambda, pre\ step, cost_rand)$;
9:	$\theta = Angle\ cost(x_{nearest}, x_{rand})$;
10:	$x_{new} = x_{nearest} + step \times [\cos(\theta)\ sin(\theta)]$;
11:	return x_{new}
12:	end function

积;Angle cost()函数功能为计算两点之间夹角值。

3 仿真实验及分析

本实验所用计算机处理器为 12th Gen Intel(R) Core (TM) i5-12500H,采用 MATLAB 2022a 进行算法仿真。设置地图大小为 500×500,扩展步长为 20,起始点由图中绿色圆圈表示;目标点由图中红色圆圈表示;所规划路径由蓝色线条绘制,障碍物用黑色块状物表示。

在同一简单障碍环境下使用 IRRT*-Connect 算法即原始算法与改进算法进行 30 次重复实验,并采用所得实验数据作为平均值,其仿真数据对比如表 5 所示。

表 5 简单环境下仿真数据表

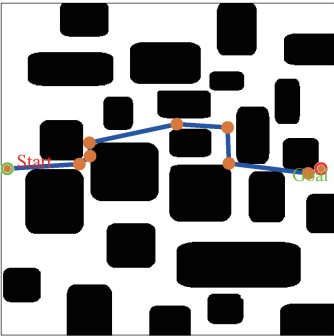
Table 5 Simulation data table in simple environment

算法	节点数目	路径长度	搜索时间
原始算法	11.77	553.59	2.39 s
改进算法	11.93	510.73	1.12 s
效果比较	+1.36%	-7.74%	-53.14%

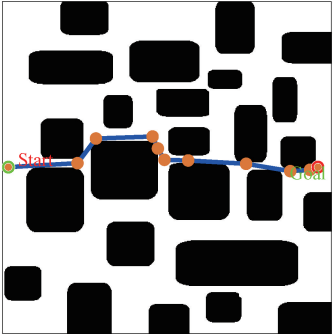
由表 5 可知,在简单障碍环境下,虽节点个数有略微增加,但其最终规划的路径长度缩短了 7.74%,同时规划时间缩短了 53.14%。表明在简单环境下该算法规划速率较快,即空旷区域规划速率加快,同时所规划路径效果更优。

采取与表 5 中数据相近的实验效果图进行对比,IRRT*-Connect 算法规划效果图如图 4(a)所示,路径代价为 552.62,改进算法规划效果图如图 4(b)图路径代价为 509.47。二者对比可直观获知所改进算法在简单障碍环境下规划效果较优。

在复杂障碍环境下使用原始算法与改进算法进行 30 次重复实验,并采用所得实验数据作为平均值,其仿真数据对比如表 6 所示。



(a) IRRT*-Connect算法
(a) IRRT*-Connect algorithm



(b)改进算法
(b) Improvement algorithm

图 4 简单环境路径规划效果

Fig. 4 Effect of path planning in simple environment

表 6 复杂环境下仿真数据表

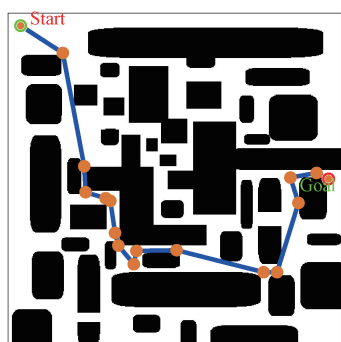
Table 6 Simulation data table in complex environments

算法	节点数目	路径长度	搜索时间
原始算法	18.76	873.50	0.65 s
改进算法	16.93	691.56	0.63 s
效果比较	-9.75%	-20.82%	-3.08%

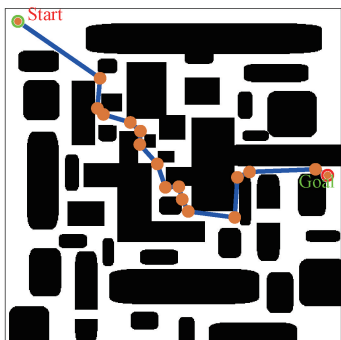
由表 6 可知,在复杂环境下,该算法路径优化效果更优,节点数目减少 9.75%,规划时间缩短 3.08%且同时路径长度缩短了 20.82%,由此可见在复杂环境下,该改进算法能够很好的适应环境,具备较强的狭窄区域规划能力,规划效果更优。

采取与表 6 中数据相近的实验效果图进行对比,IRRT*-Connect 算法规划效果图如图 5(a)所示,路径代价为 872.39,改进算法规划效果图如图 5(b)路径代价为 693.24。二者对比可直观获知在复杂障碍环境下所改进算法规划效果得到较大提升。

为验证该算法迭代优化效果,同样采用路径长度接近实验平均值的 4 次不同实验的结果进行对比,其迭代次数与路径长度关系如图 6 所示。由该图可知,该改进算法初始搜索效率较高,可快速找到初始路径进入迭代优化,且以较快速率收敛到最优路径值,且路径规划效果更优,达到预期效果。



(a) IRRT*-Connect算法
(a) IRRT*-Connect algorithm



(b) 改进算法
(b) Improvement algorithm

图5 复杂障碍环境路径规划效果

Fig. 5 Effect of path planning in complex environment

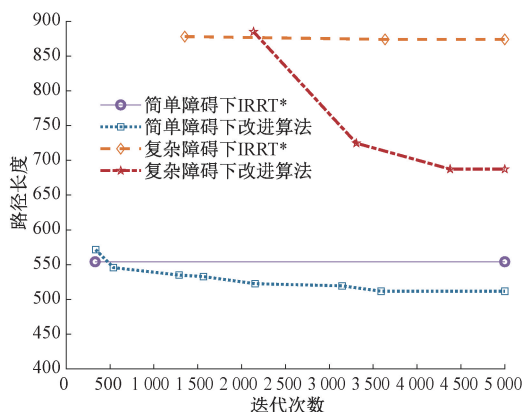


图6 不同环境下各算法路径收敛长度对比图

Fig. 6 Comparison of convergence lengths of various algorithm paths in different environments

4 结 论

针对IRRT*-Connect算法初始收敛速度较慢,适应地图能力较弱等问题,该算法以IRRT*-Connect算法作为基础算法,采用双向搜索方法获取初始解,同时提出一种采样二次约束方案以及自适应步长调整策略,使得算法初次搜索效率得到有效提高,能够根据周围环境复杂程度对步长进行自适应调整,使其能够在提高节点利用率同时较快速

过复杂障碍区域,增强其穿过狭窄区域能力,提高算法的环境适应度,规划效果更优。

参考文献

- [1] GAERTNER M, BJELONIC M, FARSHIDIAN F, et al. Collision free MPC for legged robots in static and dynamic scenes[C]. IEEE International Conference on Electronic Technology, 2021.
- [2] 孙巍伟,代锟,马飞. 空间柔性机械臂建模、控制以及轨迹规划研究综述[J]. 科学技术与工程, 2024, 24(1): 34-60.
- [3] SUN W W, DAI K, MA F. Review of modeling, control and trajectory planning of space flexible manipulators [J]. Science Technology and Engineering, 2024, 24(1): 34-60.
- [4] 何成刚,朱润智,黄挺博,等. 工业机器人轨迹规划与优化研究进展[J]. 现代制造工程, 2023, 1(7): 150-159.
- [5] HE CH G, ZHU R ZH, HUANG T B, et al. Research progress on trajectory planning and optimization of industrial robots [J]. Modern Manufacturing Engineering, 2023, 1(7): 150-159.
- [6] LI AI J, JIANG Y SH, SUN X N, et al. Research status of intelligent electric vehicle trajectory planning and its key technologies: A review[J]. Electrochem, 2022, 3(4): 688-698.
- [7] 蒋新松. 机器人学导论[M]. 沈阳: 辽宁科学技术出版社, 1994: 511-554.
- [8] JIANG X S. Introduction to robotics[M]. Shenyang: Liaoning Science and Technology Press, 1994: 511-554.
- [9] LAVALLE S M, KUFFNER J J. Randomized kinodynamic planning[J]. The International Journal of Robotics Research, 2001, 20(6): 378-400.
- [10] KUFFNER J J, LAVALLE S M. RRT-connect: An efficient approach to single query path planning[C]. IEEE International Conference on Robotics and Automation, 2000: 995-1001.
- [11] KARAMAN S, FRAZZOLI E. Sampling-based algorithms for optimal motion planning [J]. The International Journal of Robotics Research, 2011, 30(7): 846-894.
- [12] 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): 299-312.
- [13] MASHAYEHI R, IDRIS M Y, ANISI M H, et al. Hybrid RRT: A semi-dual-tree RRT-based motion planner[C]. IEEE Access, 2020, 8: 18658-18668.
- [14] 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: 115.
- [12] 杨敏豪, 张国良, 李德胜. 基于双向 F-RRT* 算法的移动机器人路径规划[J]. 电子测量技术, 2023, 46(5): 91-97.
YANG M H, ZHANG G L, LI D SH. Path planning of mobile robot based on bidirectional F-RRT* algorithm[J]. Electronic Measurement Technology, 2023, 46(5): 91-97.
- [13] 范伟伦, 李薇, 冯杭. 基于改进 RRT 算法的无人机实时航迹规划[J]. 舰船电子工程, 2019, 39(2): 56-60, 98.
FAN W L, LI W, FENG H. UAV trajectory planning based on an improved RRT algorithm [J]. Ship Electronic Engineering, 2019, 39(2): 56-60, 98.
- [14] 陈都, 侯明, 张学东. 改进 RRT 结合 B 样条的移动机器人路径规划研究[J]. 电子测量技术, 2022, 45(23): 38-44.
CHEN D, HOU M, ZHANG X D. Research on path planning of mobile robot based on improved RRT and B-spline [J]. Electronic Measurement Technology, 2022, 45(23): 38-44.
- [15] 顾子倡, 刘宇, 孙文邦, 等. 基于 RRT 的无人机动态航路规划算法[J]. 计算机科学, 2023, 50(S1): 65-69.
GU Z L, LIU Y, SUN W B, et al. UAV dynamic route planning algorithm based on RRT[J]. Computer Science, 2023, 50(S1): 65-69.
- [16] 施成龙, 行鸿彦, 王水璋, 等. 融合三步平滑与改进 RRT 算法的无人机路径规划[J]. 电子测量技术, 2023, 46(23): 43-49.
SHI CH L, XING H Y, WANG SH ZH, et al. UAV path planning combing three-step smoothing and improved RRT algorithm[J]. Electronic Measurement Technology, 2023, 46(23): 43-49.
- [17] ZHONG X Y, TIAN J, HU H SH, et al. Hybrid path planning based on safe A* algorithm and adaptive window approach for mobile robot in large-scale dynamic environment [J]. Journal of Intelligent & Robotic Systems, 2020, 99(1): 65-77.
- [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[J]. 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2014: 2997-3004. DOI:10.1109/IROS.2014.6942976.
- [19] MASHAYEKHI R, IDRIS M Y I, ANISI M H, et al. Informed RRT*-connect: An asymptotically optimal single query path planning method[C]. IEEE Access, 2020, 8: 19842-19852.

作者简介

马晓群, 硕士研究生, 主要研究方向为移动机器人路径规划算法研究。

E-mail: maxiaoqun27@163.com

王昊(通信作者), 讲师, 主要研究方向为无人机与智能无人系统。

E-mail: sirwanghao@163.com

刘磊, 教授, 主要研究方向为无人系统博弈控制、非线性系统智能控制。

E-mail: liulei@live.cn

李树, 讲师, 主要研究方向为轮式移动机器人的智能控制。

E-mail: lishu43@lnut.edu.cn