

DOI: 10.11830/ISSN.1000-5013.202207010



采用 RSPM-PS 算法的机械手 末端避障路径规划

欧阳云^{1,2}, 高振国^{1,2}, 范丽玲^{1,2}, 王继斌^{1,2}, 蒋坤良^{1,2}

(1. 华侨大学 计算机科学与技术学院, 福建 厦门 361021;
2. 华侨大学 计算机视觉与机器学习重点实验室, 福建 厦门 361021)

摘要: 为解决工业机器人机械手末端避障路径规划需要花费大量时间,且规划的路径可能会包含大量冗余分段点的问题,提出一种基于分段点迁移递归(RSPM)和递进约简(PS)的机械手末端避障路径规划方法.通过 RSPM 算法可以在碰撞段上形成分段点并进行迁移,进而递归处理各分段路径,得到一条无碰撞基础路径.通过 PS 算法可以去除基础路径上的冗余分段点,得到最终路径.结果表明:RSPM-PS 算法能快速规划更短的基础路径且能快速有效缩短路径长度.

关键词: 工业机器人; 避障路径规划; 分段点迁移递归; 递进约简; 路径优化

中图分类号: TP 242 文献标志码: A 文章编号: 1000-5013(2023)03-0290-11

Obstacle-Avoidance Path Planning of Manipulator End Using RSPM-PS Algorithm

OUYANG Yun^{1,2}, GAO Zhenguo^{1,2}, FAN Liling^{1,2},
WANG Jibin^{1,2}, JIANG Kunliang^{1,2}

(1. College of Computer Science and Technology, Huaqiao University, Xiamen 361021, China;
2. Key Laboratory of Computer Vision and Machine Learning, Huaqiao University, Xiamen 361021, China)

Abstract: It takes a lot of time to solve the obstacle avoidance path planning of industrial robot manipulator end and the planning path may contain a large number of redundant segmentation points, an obstacle-avoidance path planning method of manipulator end based on recursive segmentation point migration (RSPM) and progressive simplification (PS) is proposed. Through the RSPM algorithm, the segmentation points on the collision segment can be formed and migrated, and then the segmentation paths are recursively processed to obtain a collision-free basic path. The redundant segmentation points on the basic path can be removed by the PS algorithm to obtain the final path. The results show that the RSPM-PS algorithm can quickly plan a shorter basic path, also can quickly and effectively shorten the path length.

Keywords: industrial robot; obstacle-avoidance path planning; recursive segmentation point migration; progressive simplification; path optimization

随着机器人技术的快速发展,工业机器人已经广泛地应用于工业、农业、娱乐、医疗等许多领域^[1-5].

收稿日期: 2022-07-04

通信作者: 高振国(1976-),男,教授,博士,博士生导师,主要从事智能制造、机器视觉、无线自组网络、无线网络和网络编码的研究. E-mail: gaohit@sina.com.

基金项目: 国家自然科学基金资助项目(61972166);福建省高校产学研合作项目(2021H6030);福建省高校重点实验室基金资助项目(201910)

工业机器人的机械手/机械臂在给定空间中按特定要求进行运动, 机器人运动规划一直是工业机器人控制领域的一项核心技术. 工业生产中最常用多关节机械手, 其运动规划主要包括机械手末端路径规划和关节空间轨迹规划. 机械手末端路径规划的目的是为机械手末端寻找一条从起始点至目标点的可行路径, 关节空间轨迹规划的目的则是针对末端路径规划的各路径节点, 规划机械手各关节的运动角度或距离配置^[6], 以使机械手末端能处于特定的路径节点目标位置处.

目前, 工业机器人机械手末端路径规划算法主要包括传统路径规划算法、基于采样的路径规划算法和智能仿生路径规划算法^[7]. 传统路径规划算法主要包括模拟退火算法^[8]和人工势场法^[9], 这两种算法都易陷入局部最优解, 并且模拟退火算法缺乏对环境的普遍适应性. 基于采样的路径规划算法主要包括快速搜索随机树(RRT)^[10], 在空间较大且障碍物较多的情况下, 其随机采样的特性会导致算法效率极低, 实时性不足, 并且生成路径长度可能较长. Karaman 等^[11]对现有基于 RRT 改进的算法(RRT* 算法^[12]、RRT* -Connect 算法^[13]、Informed RRT* 算法^[14-16]、Informed RRT* -Connect 算法^[17])进行总结. 智能仿生路径规划算法主要包括蚁群算法域算法^[18]和遗传算法域算法^[19]等, 这两种算法需要选择和调整的参数较多, 参数间的耦合性较高. 在某些特殊情况下, 上述机械手末端路径规划算法路径规划空间大、空间复杂性高、运行速度慢、生成的路径长, 还经常存在大量冗余路径节点, 制约了关节空间轨迹规划的有效执行. 因此, 本文提出了一种采用分段点迁移递归-递进约简(recursive segmentation point migration-progressive simplification, RSPM-PS)算法的机械手末端路径规划方法.

1 RSPM-PS 算法的总体流程框架

给定路径规划空间场景及障碍物布局, 针对任意起始点 s 和目标点 d , 路径规划的任务就是寻找一条不与任何障碍物碰撞的路径 $path=[s, x_1, x_2, \dots, x_n, d]$, 其中, $path$ 代表从起始点 s 依次经由分段点(路径节点) x_1, x_2, \dots, x_n , 最终到达目标点 d 的分段线性路径. 各分段点之间(包括起始点 s , 目标点 d)的直连线段称为路径段.

为辅助分段点迁移递归(RSPM)算法和递进约简(PS)算法的运行, 设计 4 种基础模型: 外接矩形包围模型、碰撞检测模型、碰撞类型检测模型和碰撞规避模型. 外接矩形包围模型将任意形状空间障碍物用其外接矩形(或外接立方体)代替, 以简化空间障碍物, 从而提升碰撞检测效率. 碰撞检测模型用于检测任意两点间的直连线与空间障碍物的碰撞情况. 若与某障碍物有碰撞, 则利用碰撞类型检测模型确定碰撞类型. 根据碰撞类型和碰撞规避模型, 增加直连线分段点, 确定分段点迁移方案, 从而将分段点迁移出障碍物. 用迁移后的分段路径代替直连线, 规避当前障碍物. RSPM-PS 算法的总体流程框架, 如图 1 所示. 图 1 中: obs_1 为障碍物 1; obs_2 为障碍物 2; $n_1 \sim n_4$ 为分段点.

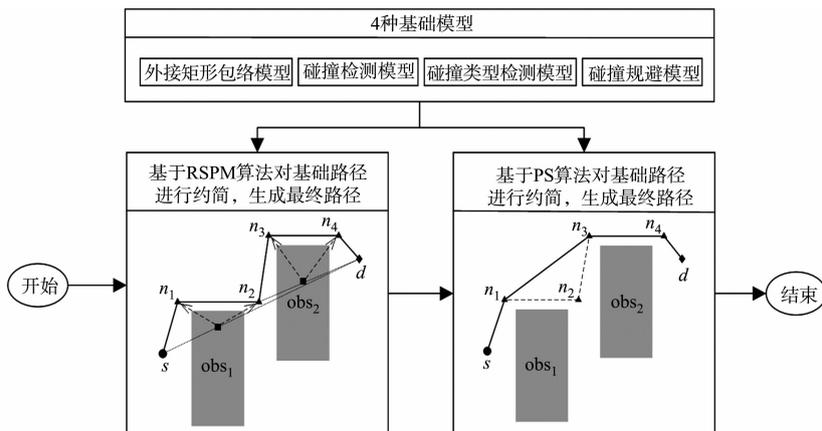


图 1 RSPM-PS 算法的总体流程框架图

Fig.1 Overall process framework of RSPM-PS algorithm

首先, 采用 RSPM 算法, 以起始点 s 至目标点 d 的直连线为初始测试路径, 根据直连线与障碍物的碰撞情况, 增加直连线的分段点, 并将分段点迁移出障碍物. 用迁移后的分段路径代替直连线, 就可规避当前障碍物. 其次, 递归检测各路径段与障碍物的碰撞情况, 必要时增加直连线的分段点, 并将分段点迁

移出障碍物,从而规避障碍物,最终得到不与障碍物碰撞的、由多段路径构成的可行规划路径(基础路径)。最后,采用PS算法,以单向递进简化的方式去除基础路径的冗余路径节点,生成简化的最终路径。

2 RSPM 算法

基础路径规划的总流程图,如图2所示。RSPM算法函数总体框架,如图3所示。采用RSPM算法规划基础路径,具体有如下4个步骤。

1) 判断两点间直线是否与障碍物发生碰撞,若不发生碰撞,则直线是一条可行路径,RSPM算法将该直线作为路径规划结果返回;若发生碰撞,则直线可能与多个障碍物发生碰撞。

2) 由于仅规避多个障碍物中离起点最近的障碍物,因此需确定该直线与相应障碍物的碰撞类型、增加的分段点及迁移方案,以避开该障碍物,这样基于直线可以得到一个避开最近障碍物的多段路径。

3) 针对新规划的各路径段,继续递归检测直连路径。

4) 由于RSPM算法规避各障碍物,使路径节点数量增大,路径段数增多,最终得到不与任何障碍物碰撞的一条可行路径,该路径就是RSPM算法要返回的基础路径。

采用RSPM算法规划的基础路径,如图4所示。图4中:粗实线表示最终的基础路径。由图4可知以下3点。

1) 对起始点 s 至目标点 d 进行碰撞检测,发现起始点 s 至目标点 d 的直线首先会和障碍物 obs_1 发生碰撞,首个碰撞段为该直线与障碍物 obs_1 的交集。

2) 由于碰撞段的两个端点在障碍物的对立边,因此,在碰撞段中点处生成两个分段点 n'_1 和 n'_2 ,并且把它们分别向左上方向、右上方向迁移至障碍物 obs_1 外,得到两个迁移后的新分段点 n_1 和 n_2 。这样起始点 s 至目标点 d 的直线变成了一个包含3段的路径 $[s, n_1, n_2, d]$ 。其中,前两个路径段都未与障碍物发生碰撞,只有第3个路径段 $[n_2, d]$ 与障碍物发生碰撞,所以前两个路径段无需继续分段。

3) 对于第3个路径段 $[n_2, d]$,则按前述过程继续处理,该路径段的首个碰撞物是障碍物 obs_2 ,且碰撞段的两个端点在障碍物的对立边,首个碰撞段的中点位于 obs_2 的上半部分。根据相应的分段点迁移方案,在碰撞段中点处生成两个分段点 n'_3 和 n'_4 ,并且把它们分别向左上方向、右上方向迁移至 obs_2 外,得到两个迁移后的新分段点 n_3 和 n_4 。这样,用路径段 $[n_2, n_3, n_4, d]$ 替换路径段 $[n_2, d]$,就避开了障碍物 obs_2 。 $[n_2, n_3, n_4, d]$ 的各路径段不再与障碍物碰撞,所以无需进一步处理,算法结束。 $[s, n_1, n_2, n_3, n_4, d]$ 就是最终的基础路径。

RSPM(起始点,目标点)的伪代码,如算法1所示。算法1返回的路径加上目标点即是基础路径。

算法1. RSPM(起始点,目标点)的伪代码。

条件:起始点、目标点、障碍物坐标。

结果:起始点至目标点的基础路径。

1) 对起始点至目标点的直线进行碰撞检测;

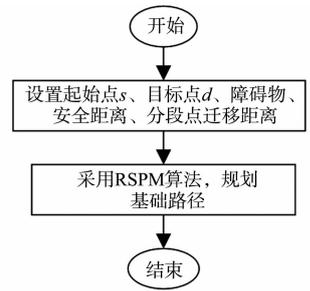


图2 基础路径规划的总流程图
Fig. 2 General flow chart of basic path planning

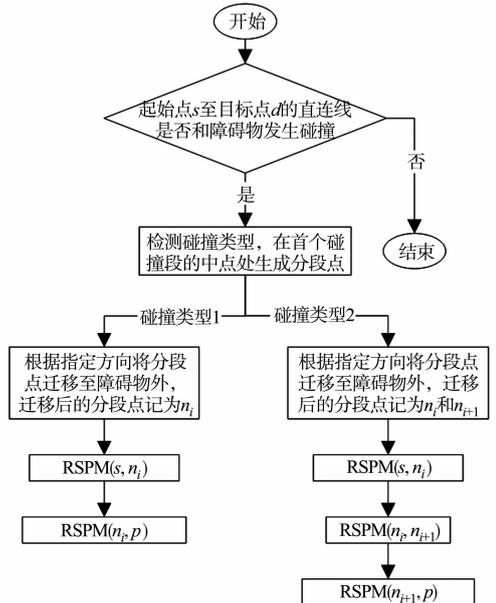


图3 RSPM算法函数总体框架
Fig. 3 RSPM algorithm function total body framework

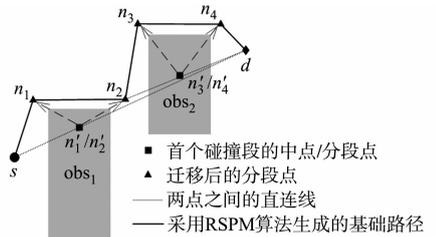


图4 采用RSPM算法规划的基础路径
Fig. 4 Basic path planning using RSPM algorithm

- 2) IF(不发生碰撞)
- 3) 将起始点加入基础路径;
- 4) RETURN;
- 5) ENDF
- 6) 根据碰撞检测得到碰撞段端点的位置,并确定碰撞类型检测;
- 7) 根据碰撞类型,在碰撞段上生成相应数量的分段点;
- 8) 根据分段点在障碍物内的位置,沿特定方向迁移出障碍物;
- 9) IF(分段点的数量为2)
- 10) RSPM(起始点,第1个分段点);
- 11) RSPM(第1个分段点,第2个分段点);
- 12) RSPM(第2个分段点,目标点);
- 13) ELSE
- 14) RSPM(起始点,分段点);
- 15) RSPM(分段点,目标点);
- 16) ENDF

3 PS 算法

若去除一条可行路径的一个分段点后,得到的路径仍然是一条不与障碍物碰撞的可行路径,则这个分段点称为原始路径上的冗余分段点.由于RSPM算法规划的基础路径可能存在冗余分段点(图4),路径 $[s, n_1, n_2, n_3, n_4, d]$ 去除分段点 n_2 后得到新路径 $[s, n_1, n_3, n_4, d]$ 仍是一条可行路径,所以 n_2 是一个冗余分段点.显然,去除冗余分段点既可以降低路径长度,又可以降低路径上分段点的数量,从而得到更优的路径.

采用PS算法检测并去除基于RSPM算法规划的基础路径的冗余分段点,从而得到最终路径,过程有如下3个步骤.

- 1) PS算法采用单向递进简化方式,按起始点 s 到终点 d 的顺序,从起始点 s 后的第2个分段点开始,依次检查各分段点至起始点 s 的直连线和障碍物是否发生碰撞,被检查的分段点称为当前分段点.
- 2) 顺次检查,直到找到某个分段点 x ,使得直连线 $[s, x]$ 与障碍物碰撞.设分段点 x 的前一个分段点为分段点 y ,则起始点 s 到分段点 y 之间的分段点为冗余分段点,删除这些冗余分段点.
- 3) 从分段点 x 的下一个分段点 z 开始,继续检查从分段点 y 到当前分段点的直连线是否与障碍物碰撞,直到到达目标点 d ,得到最终路径.

采用PS算法生成的最终路径,如图5所示.图5中:细虚线表示基于RSPM算法规划的基础路径 $[s, n_1, n_2, n_3, n_4, d]$,粗实线表示基于PS算法对该路径进行约简优化规划的最终路径 $[s, n_1, n_3, n_4, d]$.

由图5可知以下5点.

- 1) 按起始点 s 到目标点 d 的检查方向,从起始点 s 后的第2个分段点 n_2 开始检查,发现起始点 s 到分段点 n_2 的直连线与障碍物发生碰撞,可知分段点 n_1 不能删除,起始点 s 到分段点 n_1 之间没有冗余分段点.
- 2) 针对分段点 n_1 ,从分段点 n_2 的下一个分段点 n_3 开始检查,发现直连线 $[n_1, n_3]$ 不与障碍物发生碰撞,而直连线 $[n_1, n_4]$ 与障碍物发生碰撞,可知分段点 n_3 不能删除, n_1 和 n_3 之间的分段点 n_2 是冗余分段点,删除分段点 n_2 .
- 3) 针对分段点 n_3 ,从分段点 n_4 的下一个目标点 d 开始检查,发现直连线 $[n_3, d]$ 与障碍物碰撞,可知分段点 n_4 不能删除, n_3 和 n_4 之间没有冗余分段点.
- 4) 针对分段点 n_4 ,继续检查剩余各分段点.由于已检查的分段点 d 是最后一个分段点,并且 n_4 和 d 之间没有冗余分段点,所以无需进一步处理,算法结束.

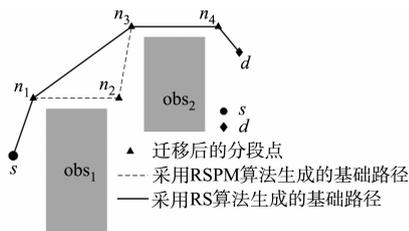


图5 采用PS算法生成的最终路径
Fig.5 Generation of final path using PS algorithm

5) 约简过程发现并删除了 1 个冗余分段点 n_2 , 得到的最终路径为 $[s, n_1, n_3, n_4, d]$.

PS 算法的伪代码, 如算法 2 所示.

算法 2. PS 算法的伪代码.

条件: 对于空间中任意的起始点和目标点之间的可行路径.

结果: 无冗余分段点的最终路径.

- 1) WHILE(TRUE)
- 2) 对起始点至目标点的直连线进行碰撞检测;
- 3) IF(不发生碰撞)
- 4) 将起始点和目标点加入最终路径;
- 5) BREAK;
- 6) ENDIF
- 7) 令 p 指向起始点后的第 2 个分段点;
- 8) WHILE(TRUE)
- 9) 对起始点至分段点 p 指向节点的直连线进行碰撞检测;
- 10) IF(发生碰撞)
- 11) 将起始点加入最终路径;
- 12) 当前分段点 p 指向的节点的前一个分段点为新的起始点;
- 13) BREAK;
- 14) ELSE
- 15) 分段点 p_4 指向路径中当前分段点 p 指向节点的后一个分段点;
- 16) ENDIF
- 17) ENDWHILE
- 18) ENDWHILE

4 基础模型

4.1 外接矩形包络模型

全局避障路径规划算法需要根据已知空间及障碍物信息, 建立二维或三维空间模型. 空间信息主要包括起始点坐标、目标点坐标和障碍物坐标. 外接矩形包络模型, 如图 6 所示.

在二维场景中, 外接矩形包络模型方案会牺牲机械手的部分自由工作空间, 但能显著降低建模难度和计算量.

4.2 碰撞检测模型

碰撞检测是路径规划最重要的步骤之一, 工业机器人机械手末端在空间中需要占据部分自由工作空间, 因此, 需要设定机械手的安全工作距离(R), 即保证机械手末端中心点与障碍物不发生碰撞的最小距离. 碰撞检测模型, 如图 7 所示. 图 7 中: 障碍物为已考虑安全工作距离的扩展后的障碍物; 距离分段点 n_1 最近的两个交点为首个碰撞段(粗实线段)的两个端点. 由图 7 可知: 分段点 n_1 至分段点 n_2 的直连线会和障碍物的边生成 4 个交点, 即分段点 n_1 至分段点 n_2 的直连线会和障碍物发生碰撞.

为判断空间中任意两点之间的直连线是否和某障碍物发生碰撞, 采用碰撞检测模型, 该模型在障碍物外接矩形包络模型的基础上, 向外扩张安全工作距离. 碰撞检测模型以扩张后的矩形作为障碍物的边

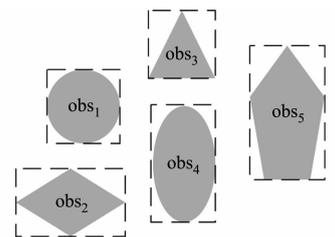


图 6 外接矩形包络模型

Fig. 6 Outer rectangular envelope model

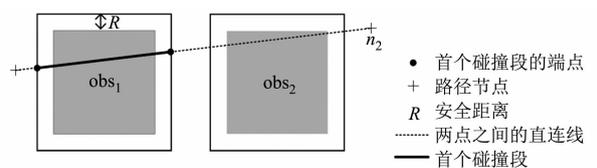


图 7 碰撞检测模型

Fig. 7 Collision detection model

为判断空间中任意两点之间的直连线是否和某障碍物发生碰撞, 采用碰撞检测模型, 该模型在障碍物外接矩形包络模型的基础上, 向外扩张安全工作距离. 碰撞检测模型以扩张后的矩形作为障碍物的边

界,当直连线跟扩展后的矩形相交的,则认为直连线与相应障碍物发生碰撞.

在任意两点之间直连线的碰撞检测中,只需要判断该直连线是否与障碍物碰撞,而不需要判断障碍物的数量. 后续规避处理总是针对与起始点距离最近的障碍物,而对于其他后发生碰撞的障碍物,则是在本次规避处理得到的新路径段的基础上,通过新路径段的碰撞检测和碰撞规避进行解决.

4.3 碰撞类型检测模型

当两分段点之间的直连线和障碍物发生碰撞时,需要进行碰撞避障处理,但规避方案依赖于碰撞避障类型. 根据首个碰撞段的两个端点相对于障碍物的位置,把碰撞类型检测模型分为两类,如图 8 所示. 图 8 中:模型 1 碰撞段的两个端点在障碍物的相邻边;模型 2 碰撞段的两个端点在障碍物的对立面.

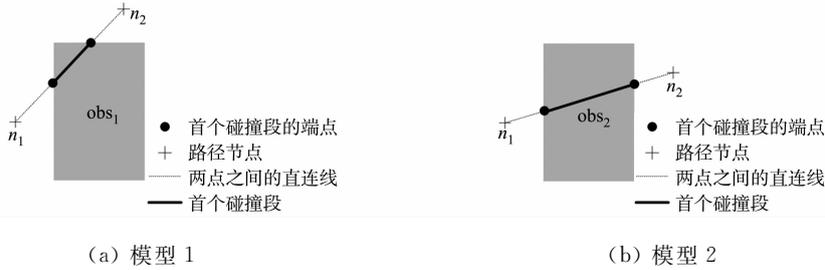


图 8 碰撞类型检测模型

Fig. 8 Collision type detection model

4.4 碰撞规避模型

碰撞类型检测模型 1 在首个碰撞段的中点处生成 1 个分段点,将分段点向碰撞段第 1 个端点所在边和碰撞段第 2 个端点所在边的交点方向迁移至障碍物外,得到迁移后的新位置. 碰撞类型检测模型 1 的碰撞规避模型 1,如图 9(a)所示. 由图 9(a)可知:碰撞段的第 1 个端点位于障碍物的左边,第 2 个端点位于障碍物的上边,因此,将分段点向左上方交点方向迁移至障碍物外,得到分段点迁移后的位置.

碰撞类型检测模型 2 在首个碰撞段的中点检测障碍物中的相对位置,若该中点位于障碍物内部的上方,则在碰撞段中点处生成两个分段点,将其中一个分段点向碰撞段第 1 个端点所在边与障碍物所在边的交点方向迁移至障碍物外,另一个分段点则向碰撞段第 2 个端点所在边和障碍物所在边的交点方向迁移至障碍物外,得到两个分段点迁移后的新位置. 其他情况做类似处理. 碰撞类型检测模型 2 的碰撞规避模型 2,如图 9(b)所示. 图 9(b)中:碰撞段的中点位于障碍物内部的上方,并且碰撞段的第 1 个端点位于障碍物的左边,第 2 个端点位于障碍物的右边,因此,将两个分段点依次分别向左上方交点方向、右上方交点方向迁移至障碍物外.

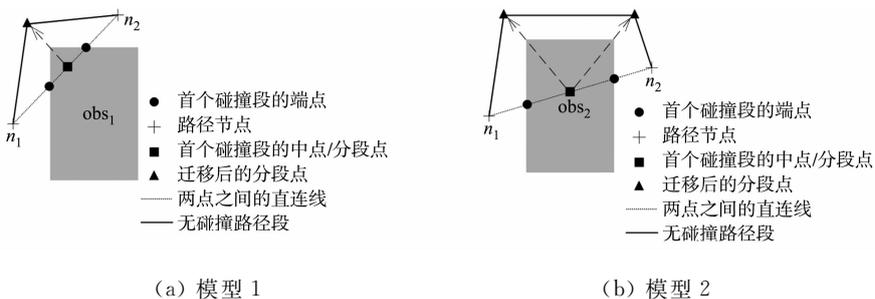


图 9 碰撞规避模型

Fig. 9 Collision avoidance model

5 实验仿真与结果分析

通过仿真实验对比测试 RSPM-PS 算法的性能. 采用 MATLAB 2016a 软件,系统配置为 64 位 Windows 操作系统,硬盘为 10 GB 以上的 Intel(R) Core(TM) 2.50 GHz CPU,4 GB RAM.

4 种代表性路径规划场景分别为单障碍物场景、狭窄通道场景、多障碍物二维场景和多障碍物三维场景. 评价指标为平均运行时间和平均路径长度. 4 种代表性路径规划算法分别为 RRT*^[12] 算法、RRT*-Connect^[13] 算法、Informed RRT*^[14-16] 算法、Informed RRT*-Connect^[17] 算法. 对 5 种算法

(RSPM 算法与 4 种代表性路径规划算法)及进行 100 次实验,以各评价指标值的平均值为最终指标值,并计算其 95% 的置信区间。

实验仿真的参数设置,如表 1 所示.表 1 中: N 为最大迁移点生成数; L_s 为生长步长; r 为到达目标点区域半径; M 为最大树节点生成数。

表 1 实验仿真的参数设置

Tab.1 Parameters setting of experimental simulation

参数	单障碍物场景	狭窄通道场景	多障碍物二维场景	多障碍物三维场景
R/mm	1.0	0.5	3.0	3.0
$N/\text{个}$	400	1 200	1 200	1 600
L_s/mm	5	5	10	10
r/mm	5	5	10	10
$M/\text{个}$	400	1 200	1 200	1 600

5.1 单障碍物场景

基于文献[17],在仅有 1 个障碍物的二维空间中,为了测试 RSPM 算法在不同起始点至目标点距离的性能,设立单障碍物场景.单障碍物场景,如图 10 所示.图 10 中:5 条路径分别是 5 种算法在起始点至目标点距离为 100 mm 时的一次路径规划结果; X 为横坐标值; Y 为纵坐标值。

单障碍物场景下的实验结果,如图 11 所示.图 11 中: \bar{t}_{run} 为平均运行时间; \bar{L}_{pat} 为平均路径长度; \bar{L}_{pat} 为起始点 s 到目标点 d 的平均路径长度。

由图 11(a)可知:5 组不同起始点 s 到目标点 d 距离下 RSPM 算法规划路径的平均运行时间分别为 0.001 5,0.002 8,0.002 7,0.002 7,0.002 6 s,而对比算法中表现最好的 Informed RRT* -Connect 算法规划路径的平均运行时间分别为 0.080 8,0.286 5,0.333 8,0.272 2,0.319 2 s;RSPM 算法规划路径的最小运行时间为 0.001 1 s,而对比算法规划路径的最小运行时间为 0.053 8 s。

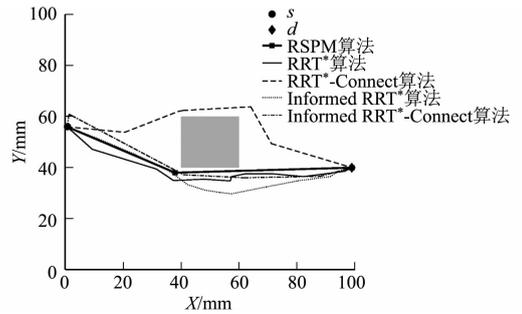
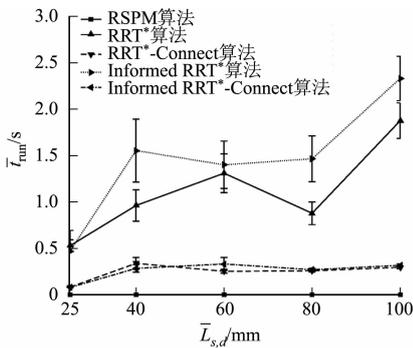
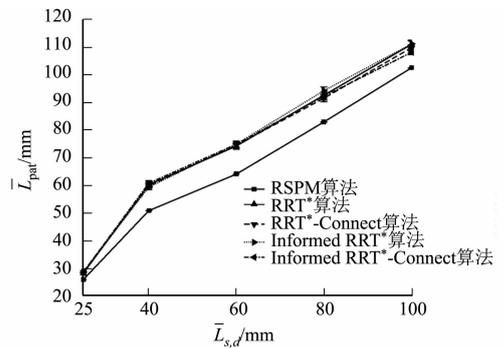


图 10 单障碍物场景

Fig.10 Single obstacle scene



(a) 平均运行时间



(b) 平均路径长度

图 11 单障碍物场景下实验结果

Fig.11 Experimental results of single obstacle scene

由图 11(b)可知:5 组不同起始点 s 到目标点 d 距离下 RSPM 算法规划路径的平均路径长度分别为 25.999 8,50.954 1,64.164 1,82.995 1,102.546 9 mm,而对比算法中表现最好的 Informed RRT* -Connect 算法规划路径的平均路径长度分别为 28.626 5,60.881 1,74.936 7,92.302 6,108.129 6 mm。

5.2 狭窄通道场景

基于文献[17],为了测试 RSPM 算法在不同宽度狭窄通道下的性能,设立狭窄通道场景进行仿真实验,5 组狭窄通道宽度分别为 2,3,4,5,6 mm.狭窄通道场景,如图 12 所示.图 12 中:5 条路径分别是 5 种算法在狭窄通道宽度为 6 mm 时的一次路径规划结果.狭窄通道场景下的实验结果,如图 13 所示.图 13 中: W 为狭窄通道宽度。

由图 13(a)可知:5 组不同狭窄通道宽度下 RSPM 算法规划路径的平均运行时间分别为 0.008 4,0.008 1,0.008 3,0.008 4,0.008 6 s,而对比算法中表现最好的 Informed RRT*-Connect 算法规划路径的平均运行时间分别为 9.985 0,9.688 7,6.328 3,3.813 8,3.228 4 s;RSPM 算法规划路径的最小运行时间为 0.006 7 s,而对比算法规划路径的最小运行时间为 1.001 6 s.

由图 13(b)可知:5 组不同狭窄通道宽度下 RSPM 算法规划路径的平均路径长度分别为 220.609 0,217.734 2,214.863 4,211.997 1,209.135 4 mm,对比

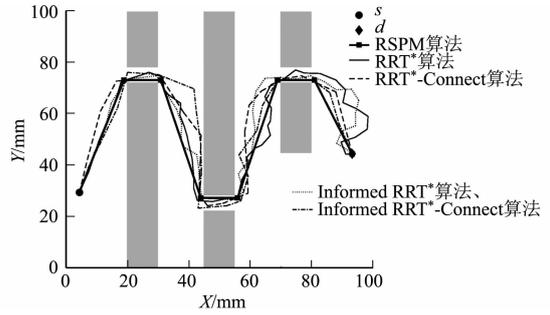
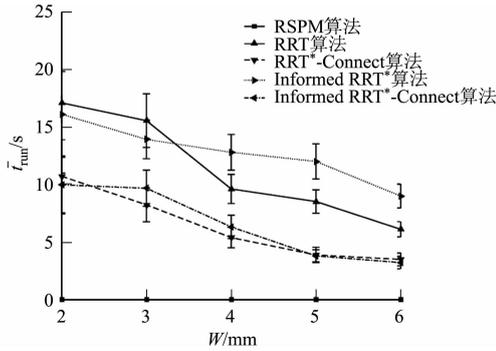
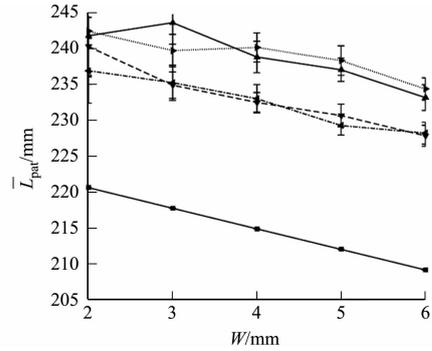


图 12 狭窄通道场景

Fig. 12 Narrow channel scene



(a) 平均运行时间



(b) 平均路径长度

图 13 狭窄通道场景下的实验结果

Fig. 13 Experimental results of narrow channel scene

算法中表现最好的 Informed RRT*-Connect 算法生成路径的平均路径长度分别为 236.922 2,235.252 7,233.014 4,229.275 9,228.219 2 mm.

5.3 多障碍物二维场景

为了测试 RSPM 算法在不同起始点至目标点距离下的性能,设立多障碍物二维场景进行仿真实验,5 组起始点至目标点坐标距离分别是 200,300,400,500,600 mm.多障碍物二维场景,如图 14 所示.图 14 中:5 条路径分别是 5 种算法在起始点至目标点距离为 600 mm 时的一次路径规划结果.

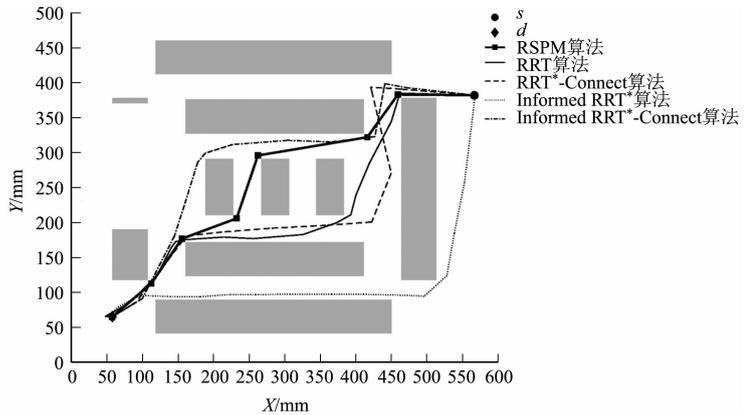


图 14 多障碍物二维场景

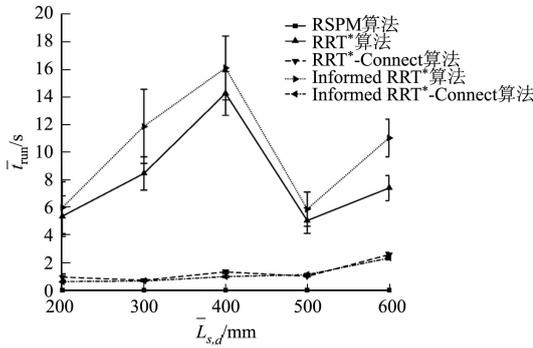
Fig. 14 Multi-obstacle 2D scene

多障碍物二维场景下的实验结果,如图 15 所示.由图 15(a)可知:5 组不同

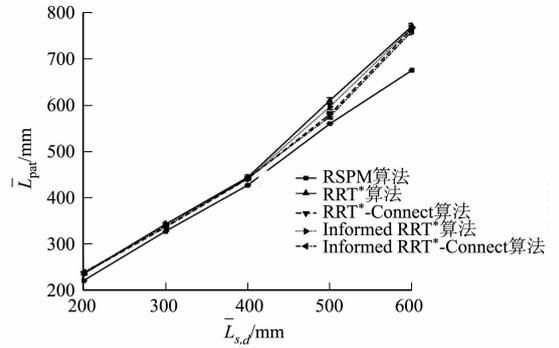
起始点 s 到目标点 d 距离下 RSPM 算法规划路径的平均运行时间分别为 0.004 9,0.003 1,0.004 8,0.010 5,0.014 3 s,而对比算法中表现最好的 Informed RRT*-Connect 算法规划路径的平均运行时间分别为 0.636 7,0.677 9,0.994 3,1.136 9,2.331 1 s;5 组不同起始点 s 到目标点 d 下 RSPM 算法规划路径的最小运行时间仅为 0.002 5 s,而 4 种对比算法规划路径的最小运行时间为 0.237 0 s.由图 15 (b)可知:5 组不同起始点 s 到目标点 d 距离下 RSPM 算法规划路径的平均路径长度分别为 221.092 3,327.672 2,426.961 4,559.542 5,675.023 2 mm,而对比算法中表现最好的 Informed RRT*-Connect 算法规划路径的平均路径长度分别为 238.614 4,338.274 9,441.838 1,574.968 1,759.370 4 mm.

5.4 多障碍物三维场景

在具有多个障碍物的三维场景中,为了测试 RSPM 算法在不同起始点至目标点距离下的性能,设



(a) 平均运行时间



(b) 平均路径长度

图 15 多障碍物二维场景下实验结果

Fig. 15 Experimental results of multi-obstacle 2D scene

立多障碍物三维场景进行仿真实验,5组起始点和目标点坐标的距离分别是200,300,400,500,600 mm.多障碍物三维场景,如图16所示.图16中:5条路径分别是5种算法在起始点至目标点距离为600 mm时的一次路径规划结果,Z为Z轴坐标值.

此外,可以根据现实情况设置场景边界.在该实验中,假设Z轴0刻度为地面,即5种算法在Z轴0刻度及以下区域不规划路径.

多障碍物三维场景下实验结果,如图17所示.

由图17(a)可知:5组不同起始点至目标点距离下RSPM算法规划路径的平均运行时间分别为

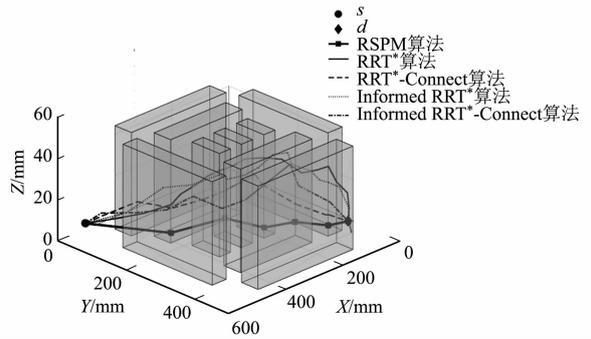
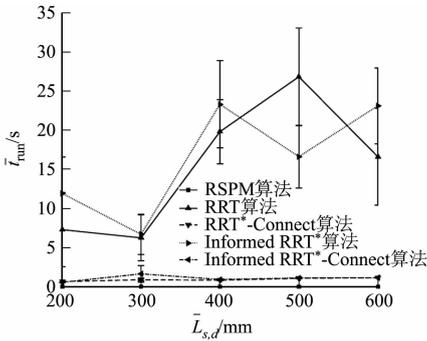
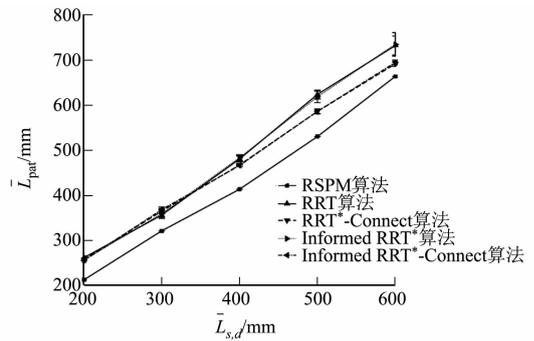


图 16 多障碍物三维场景

Fig. 16 Multi-obstacle 3D scene



(a) 平均运行时间



(b) 平均路径长度

图 17 多障碍物三维场景下实验结果

Fig. 17 Experimental results of multi-obstacle 3D scene

0.008 5,0.011 7,0.008 0,0.015 7,0.019 5 s,而对比算法中表现最好的RRT*-Connect算法规划路径的平均运行时间分别为0.673 9,0.917 0,0.856 7,1.095 4,1.183 7 s;5组不同起始点和目标点距离下RSPM算法生成路径的最小运行时间仅为0.005 4 s,而4种对比算法规划路径的最小运行时间为0.193 3 s.

由图17(b)可知:5组不同起始点至目标点距离下RSPM算法规划路径的平均路径长度分别为212.578 4,321.458 8,413.838 8,530.769 7,664.367 4 mm,而对比算法中表现最好的RRT*-Connect算法生成路径的平均路径长度分别为257.719 0,365.224 9,467.727 1,586.573 4,692.050 4 mm.

5.5 路径约简

为了测试PS算法约简路径的性能,在多障碍物三维场景中,基于RSPM算法和4种对比算法在任意起始点s到目标点d之间规划一条路径,再利用PS算法对生成的路径进行约简.路径约简实验结果,如图18所示.图18中: N_p 为分段点数.

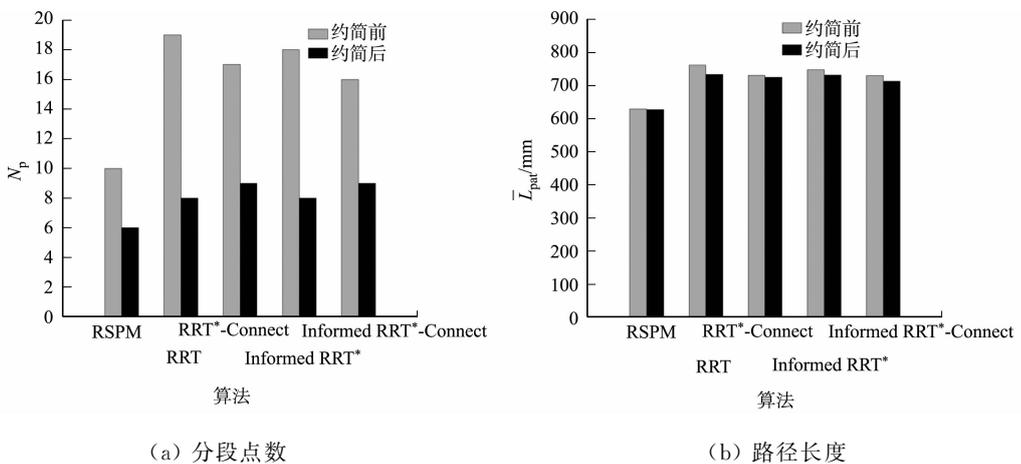


图18 路径约简实验结果

Fig. 18 Experimental results of road simplification

由图18可知:5种算法规划路径的分段点数比约简前分别减少了40%,57%,47%,55%,43%;路径长度分别减少了0.39%,3.74%,0.75%,2.07%,2.30%;5次路径约简所用的时间分别为0.0045,0.0101,0.0040,0.0076,0.0055s。

6 结论

1) 提出了一种采用分段点迁移递归(RSPM)和递进约简(PS)的机器人末端避障路径规划算法(RSPM-PS),其中,RSPM算法用于基础路径规划,PS算法用于约简基础路径,生成最终路径,是一种通用的路径优化处理方法。

2) 在4种仿真场景中,将RSPM算法和4种对比算法进行对比实验,RSPM算法在平均运行时间和平均路径长度两个性能指标上均优于4种对比算法,结果证明了RSPM算法的有效性和高效性。

3) 利用PS算法对5种算法生成的路径进行约简,可以快速去除路径上的冗余分段点,进一步缩短了平均路径长度,路径约简结果证明了PS算法的有效性和高效性。

在未来的工作中,RSPM-PS算法将被应用于工业机器人机械手末端避障路径规划,而路径规划结果将被应用于工业机器人机械手在关节空间中的避障轨迹规划。

参考文献:

- [1] BEN-ARI M, MONDADA F. Robots and their applications[M]. Berlin: Springer, 2018.
- [2] MOSADEGHZAD M, KALYAN D, KALIYANUROV Z, *et al.* Towards enhancing modular production systems by integrating a collaborative robotic manipulator[C]// IEEE International Conference on Mechatronics and Automation. Tianjin: IEEE Press, 2019: 1750-1755. DOI:10.1109/ICMA.2019.8816444.
- [3] YAN Bin, PAN Fan, LEI Xiaoyan, *et al.* A real-time apple targets detection method for picking robot based on improved YOLOv5[J]. Remote Sensing, 2021, 13(9): 1619. DOI:10.3390/rs13091619.
- [4] DUCUTJ, VALENZUELA I, DADIOS E, *et al.* Design and development of robotic arm movements and body frame for a social robot for graduation rites[C]// IEEE 13th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management. Manila: IEEE Press, 2021: 1-6. DOI:10.1109/HNICEM54116.2021.9731930.
- [5] ZHAO Fengda, YANG Zhikai, LI Xianshan, *et al.* Extract executable action sequences from natural language instructions based on DQN for medical service robots[J]. International Journal of Computers, Communications and Control, 2021, 16(2): 1-12. DOI:10.15837/ijccc.2021.2.4115.
- [6] SEPEHRI A, MOGHADDAM A. A motion planning algorithm for redundant manipulators using rapidly exploring randomized trees and artificial potential fields[J]. IEEE Access, 2021, 9: 26059-26070. DOI: 10.1109/ACCESS.2021.3056397.
- [7] 张广林, 胡小梅, 柴剑飞, 等. 路径规划算法及其应用综述[J]. 现代机械, 2011, 5(14): 85-90. DOI:10.13667/j.cnki.

52-1046/th. 2011. 05. 014.

- [8] 陶重彝,雷祝兵,李春光,等. 基于改进模拟退火算法的搬运机器人路径规划[J]. 计算机测量与控制, 2018, 26(7): 182-185. DOI:10.16526/j.cnki.11-4762/tp.2018.07.040.
- [9] GAI Shengnan, SUN Rui, CHEN Shunjun, *et al.* 6-DOF robotic obstacle avoidance path planning based on artificial potential field method[C]// 16th International Conference on Ubiquitous Robots. Jeju: IEEE Press, 2019: 165-168. DOI:10.1109/URAI.2019.8768792.
- [10] QIN Zekui, WANG Rui, DONG Xiwang, *et al.* Three-dimensional path planning for unmanned aerial vehicles based on the developed RRT algorithm[C]// IEEE CSAA Guidance, Navigation and Control Conference. Xiamen: IEEE Press, 2018: 1-5. DOI:10.1109/GNCC42960.2018.9018965.
- [11] KARAMAN S, FRAZZOLI E. Sampling-based algorithms for optimal motion planning[J]. The International Journal of Robotics Research, 2011, 30(7): 846-894. DOI:10.1177/0278364911406761.
- [12] 李剑光,刘椿鹏,周梓燚,等. 基于改进 RRT* 算法的机械手路径规划[J]. 组合机床与自动化加工技术, 2022, 3(3): 9-12. DOI:10.13462/j.cnki.mmtamt.2022.03.003.
- [13] KUFFNER J, LAVALLE S. RRT-connect: An efficient approach to single-query path planning[C]// IEEE International Conference on Robotics and Automation. San Francisco: IEEE Press, 2000: 995-1001. DOI:10.1109/ROBOT.2000.844730.
- [14] KLEMM S, OBERLÄN J, HERMANN A, *et al.* RRT* -connect: Faster, asymptotically optimal motion planning [C]// IEEE International Conference on Robotics and Biomimetics. San Francisco: IEEE Press, 2015: 1670-1677. DOI:10.1109/ROBIO.2015.7419012.
- [15] GAMMELL J, SRINIVASA S, BARFOOT T. Informed RRT: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic[C]// IEEE/RSJ International Conference on Intelligent Robots and Systems. Zhuhai: IEEE Press, 2014: 2997-3004. DOI:10.1109/IROS.2014.6942976.
- [16] GAMMELL J, BARFOOT T, SRINIVASA S. Informed sampling for asymptotically optimal path planning[J]. IEEE Transactions on Robotics, 2018, 34(4): 966-984. DOI:10.1109/TRO.2018.2830331.
- [17] MASHAYEKHI R, IDRIS M, ANISI M, *et al.* Informed RRT* -connect: An asymptotically optimal single-query path planning method[J]. IEEE Access, 2020, 8: 19842-19852. DOI:10.1109/ACCESS.2020.2969316.
- [18] ZHAO Huadong, LEI Chaofan, JIANG Nan. A path planning method of robot arm obstacle avoidance based on dynamic recursive ant colony algorithm[C]// IEEE International Conference on Power, Intelligent Computing and Systems. Shenyang: IEEE Press, 2019: 549-552. DOI:10.1109/ICPICS47731.2019.8942495.
- [19] 韩涛,吴怀宇,杜钊君,等. 基于遗传算法的机械手实时避障路径规划研究[J]. 计算机应用研究, 2013, 30(5): 1373-1376. DOI:10.3969/j.issn.1001-3695.2013.05.023.

(责任编辑: 陈志贤 英文审校: 吴逢铁)