

基于改进RRT*算法的机械臂路径规划研究

刘学深^{1a}, 曹立佳^{1a,1b,2}

(1. 四川轻化工大学 a. 自动化与信息工程学院, b. 计算机科学与工程学院, 四川 宜宾 644000;

2. 企业信息化与物联网测控技术四川省高等学校重点实验室, 四川 宜宾 644000)

摘要: RRT(Rapidly exploring Random Tree)是一种基于采样的路径规划算法,非常适用于机器人的路径规划中,但是传统RRT*算法存在耗时长、占用内存较大等缺点。所以针对这些问题提出一种改进RRT*算法,该算法优化了父节点选取范围,在传统随机采样机制的基础上引入了目标偏置采样和启发式策略,减少了算法耗时且缩短了路径长度;引入了节点拒绝策略,消除转弯角太大的冗余路径的同时也进一步提升了算法效率。利用MATLAB进行了仿真实验验证,结果表明改进RRT*算法能在更短的时间内搜索到一条从起点到终点的最短无碰路径,并且可以很好地应用于机械臂的路径规划中。

关键词: 路径规划;改进RRT*;目标偏置采样;启发式策略;节点拒绝策略;机械臂

中图分类号: TP242.2

文献标志码: A

引言

随着我国生产业从中国制造向中国智能制造转型,我国对工业机械臂提出了更高程度智能化与精度的要求。如何实现工业机械臂在有障碍物的狭小环境中高效安全地工作,成为机械臂研究中极其重要的课题。快速扩展随机树(Rapidly Expanding Random Tree, RRT)算法具有概率完备的特点,且无需对系统进行建模,可直接应用于非完整约束系统的规划,其算法复杂度小,十分适合解决高维空间中的避障规划问题^[1],经常被人们应用

于机械臂、移动机器人和无人机等路径规划领域中。之后出现的RRT*算法能收敛到全局最优解,但是它往往需要很高的迭代次数^[2]。目前已有众多学者对RRT*算法做出了一系列改进,且应用在了机械臂的路径规划中。

文献[3]针对RRT*算法随机性大和效率低的缺点,加入了自适应步长策略和节点拒绝策略,提高了RRT*搜索速度,但是该算法易陷入局部最优。文献[4]提出了节点扩展能力检测策略,减少了无效节点的生成,但存在计算量太大的缺陷。文献[5]加入了目标引力;文献[6]提出了柯西目标引力和节点

收稿日期:2022-11-23

基金项目:四川省科技创新苗子工程项目(MZGC20230084);中国高校产学研创新基金项目(2021ZYA11002);四川轻化工大学研究生创新基金项目(y2021088)

通信作者:曹立佳(1982-),男,副教授,博士,研究方向为无人系统导航与控制,(E-mail)caolj@suse.edu.cn

拒绝策略,能够剔除不必要的采样节点;文献[7]提出了动态采样域策略,提高了拓展方向的确切性;文献[8]通过构建李势场改进了节点搜索方向及步长;文献[9]将启发函数应用在双向RRT采样过程中;文献[10]增加了目标偏置策略;文献[11]改变了采样点生成机制,通过建立采样池来对采样点进行优选;文献[12]提出了高斯采样的方法,增加了对障碍物附近区域的采样次数;文献[13]采用了覆盖剔除机制,减少了重复采样;文献[14]采用了椭圆子集约束采样,提高了算法收敛速度。

可以看出以上文献都是从以下3种方式中选取1~2种方式去提升算法效率:(a)改变扩展方向(设计思想:为了直接减小耗时);(b)删除冗余节点(设计思想:为了减小内存,间接减小耗时);(c)改变步长

(设计思想:为了优化路径,进而加快收敛)。这三种方式都是从不同的维度去减小算法耗时,但是以上文献算法要么计算量太大不适用于复杂环境,要么耗时减小得不够明显,而且在复杂环境中单独采取一种方式的效果往往并不是最优的,因此本文将联合以上3种方式,从不同维度出发,提出一种方法简明、占用内存小且耗时短的改进RRT*算法,以提升算法综合性能。

1 机械臂运动学模型的建立

以某品牌的6轴串联机械臂为研究对象,首先运用DH参数法建立6轴机械臂运动学模型。该方法是在每个连杆上建立一个坐标系,通过齐次变换来实现两个连杆上的坐标变换^[15],模型如图1所示。

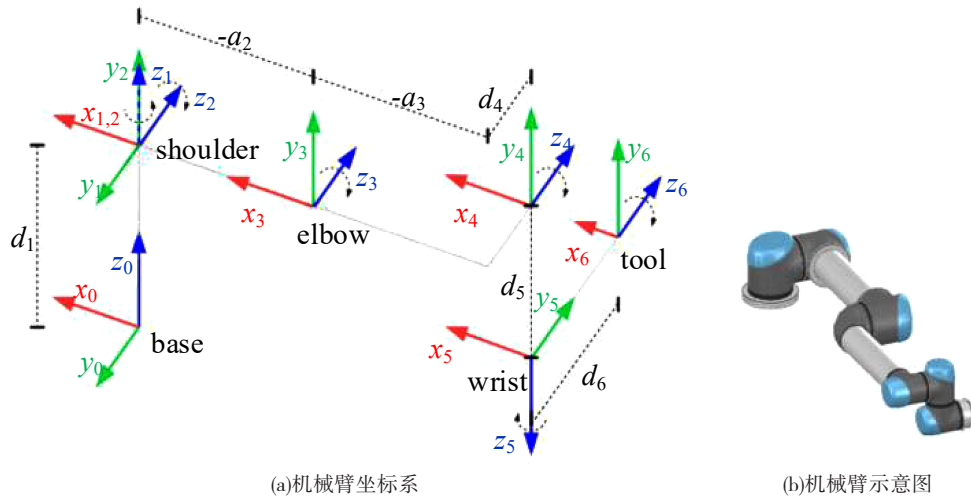


图1 机械臂坐标系

DH法的4个参数可以表达相邻两个关节之间的相对位置,进而可以构建出6轴机械臂运动学模型。图1中 $i(i=1,2,\dots,6)$ 表示连杆, a_{i-1} 表示转轴 i 和转轴 $(i-1)$ 之间的连杆长度; d_i 表示转轴 i 的连杆偏距,即 a_{i-1} 与 a_i 两条直线与转轴 i 交点的距离; α_{i-1} 表示转轴 $(i-1)$ 相对于转轴 i 的连杆转角, θ_i 表示关

节角度。

对全部连杆规定坐标系后,由两个旋转和两个平移来建立相邻两连杆 $(i-1)$ 与 i 之间的相对关系,且可由4个齐次变换来描述,可表示为矩阵 T^{i-1} ^[16-17],如式(1)所示:

$${}^{i-1}T = \begin{bmatrix} \cos \theta_i & -\sin \theta_i & 0 & a_{i-1} \\ \sin \theta_i \cos \alpha_{i-1} & \cos \theta_i \cos \alpha_{i-1} & -\sin \alpha_{i-1} & -d_i \sin \alpha_{i-1} \\ \sin \theta_i \sin \alpha_{i-1} & \cos \theta_i \sin \alpha_{i-1} & \cos \alpha_{i-1} & d_i \cos \alpha_{i-1} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

RRT*算法应用在机械臂路径规划的流程如图2所示。首先输入机械臂的初始状态,然后经过采样和生成新节点的过程,再进行碰撞检测,然后

再进行父节点重选和路径剪枝过程对路径进行优化,以减小路径代价。重复以上的流程,当达到最大的迭代次数时,最终输出路径点信息。

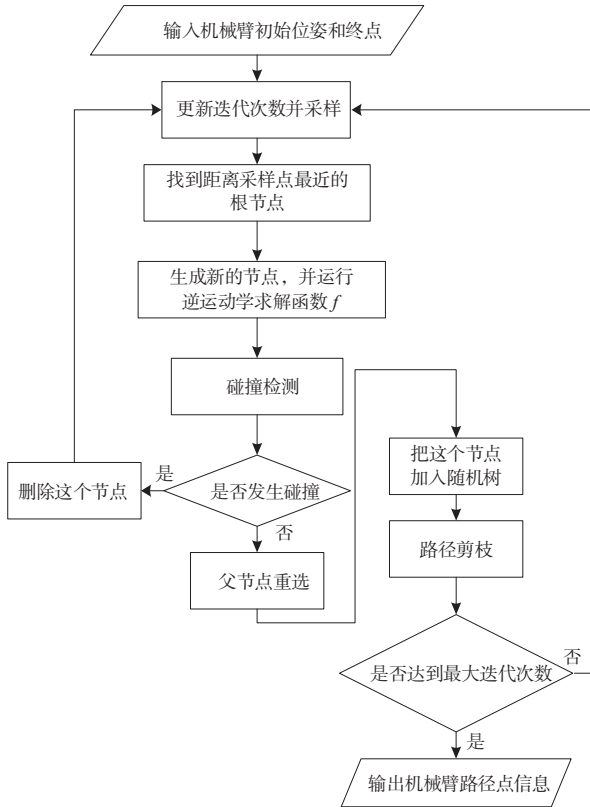


图2 RRT*机械臂路径规划算法流程

2 RRT*算法原理

RRT是一种基于采样的路径规划算法,该算法具有概率完备性。首先在搜索空间中随机采样一点,进而确定扩展方向,然后向该方向以某一固定步长生成新节点,重复这些操作直到节点靠近目标点。该算法搜索能力强,只要不限运行时间和迭代次数,总能在存在障碍物的复杂空间中找到一条无碰撞的可行路径^[18]。

为了解决 RRT 算法规划出来的路径不是最优的问题,RRT*在原始 RRT 算法每次迭代生成新节点后加入了父节点重选^[19]和路径剪枝的过程,然后不断迭代,最终可以让规划路径趋于渐进最优。其中 RRT*父节点重选的过程如图 3 所示,图中小圆圈中的数字是节点编号,而直线中间的数字代表路径代价。

图 3 中 1 号点为 X_{ini} ,首先由传统 RRT 算法生成的 6 号节点 X_{new} 的父节点为 3 号节点,该路径为 1-2-3-6,整体路径代价为 $5+4+4=13$ 。然后 RRT*算法

将以 6 号节点为中心以自己定义的半径范围内找出“近邻”节点 2、3、4、5,并且依次与 6 号节点相连计算出相应的路径代价,如果路径代价小于原来的 13,则替换 6 号节点的父节点 3。

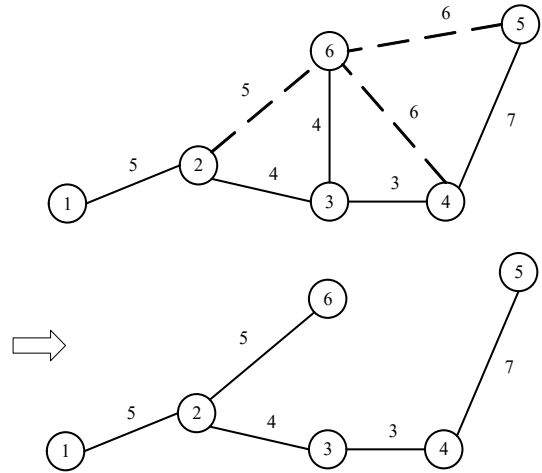


图3 父节点重选

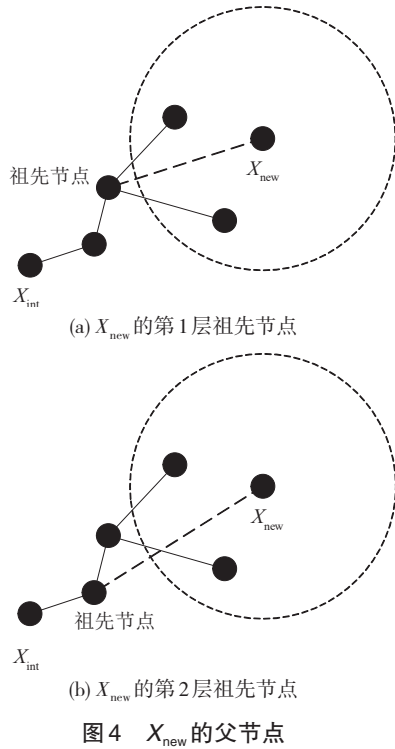
6 号节点的“近邻”节点有 2、4、5,选择 4 号节点为父节点的路径为 1-2-3-4-6,路径代价为 $5+4+3+6=18$,由于 $18 > 13$,所以不替换原来的父节点;选择 2 号节点为父节点的路径为 1-2-6,路径代价为 $5+5=10$,由于 $10 < 13$,显然这条路径更好,所以替换 6 号节点的父节点 3 号点为 2 号节点,路径长度明显缩短。

尽管 RRT*优化了搜索路径,但是 RRT*规划出的路径质量与迭代次数有关,往往需要设定较大的迭代次数才能搜索出最优路径,所以为提高 RRT*的算法效率,本文算法进行改进。

3 RRT*算法改进

3.1 改进近邻节点选取范围

由于 RRT*是在定义的半径内寻找近邻节点,使得剪枝和父节点重连策略的优化程度不充分,算法效率低^[20]。针对这些问题,本文将扩大近邻节点的选取方式。具体思路是考虑缩短从起点到 X_{new} 的路径长度,找出 X_{new} 第 n 层的祖先节点 O_2 和第 $n+1$ 层的祖先节点 O_3 与原本近邻节点 O_1 组成一个新的近邻节点集合 A ,然后依次与 X_{new} 相连进行比较,从中选取使得 X_{new} 路径代价最小的节点作为 X_{new} 的父节点,如图 4(以 $n = 1$ 为例)所示。



重新选择父节点后,路径将会得到缩短,求得各个集合分别为:

$$O_1 = \{X_{near}\};$$

$$O_2 = \{X_{new} \text{ 的第 } n \text{ 层的祖先节点}\};$$

$$O_3 = \{X_{new} \text{ 的第 } n + 1 \text{ 层的祖先节点}\}.$$

3.2 全局采样方式优化

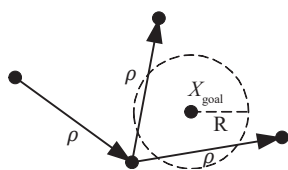
由于RRT*是在整个搜索空间内随机均匀采样,随机性大,为了减小冗余搜索,缩短路径搜索时间^[21],将缩小采样空间:

1)以一定概率 a 在搜索空间中均匀随机采样;

2)以一定概率 b 采取目标偏置采样,令 $X_{rand} =$

X_{goal} ;

3)如图5所示,当生成新节点坐标 X_{new} 时,会判断 X_{new} 是否处于目标点 X_{goal} 的邻域内。如果是,则将 X_{new} 作为 X_{goal} 的父节点,但是步长 ρ 可能会设置得比 X_{goal} 邻域半径 R 大,导致随机树在 X_{goal} 扩展时会发生振荡,导致算法收敛速度降低。



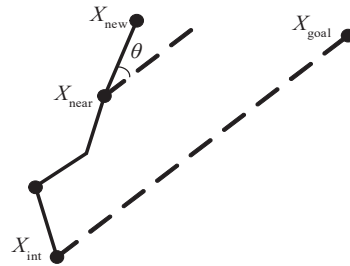
为了避免这种情况,更快地得到最优路径,所以将以一定概率 c ,不进行采样,而是遍历所有随机树节点,找出代价 $Cost$ 最小的节点 X_k 如式(2)所示:

$$Cost = C^k + d(X_k, X_{goal}) \quad (2)$$

其中, C^k 为从起点 X_{int} 到 X_k 的实际路径代价大小, $d(X_k, X_{goal})$ 为节点 X_k 到目标点 X_{goal} 的欧式距离。连接 X_k 和 X_{goal} ,对 X_k 到 X_{goal} 的这段路径进行碰撞检测,如果没发生碰撞,则直接将 X_k 作为 X_{goal} 的父节点,把 X_{goal} 加入随机树中,生成路径;如果发生碰撞,则改为均匀随机采样,其中 $a + b + c = 1$ 。

3.3 节点拒绝策略

为了避免随机树向没必要探索的空间扩展,且满足机器人运动学约束,本文将引入节点拒绝策略,具体方法如图6所示。在每一次随机树扩展生成新节点 X_{new} 时,判断 X_{near} 与 X_{new} 的连线与起点 X_{int} 和目标点 X_{goal} 的连线夹角是否大于某一个角度值,如果夹角太大,大于 σ (本文设置为 60°),则删除 X_{new} , θ 的计算公式如式(3)所示。



$$\theta = \arccos \frac{(\overrightarrow{X_{new} - X_{near}}) \cdot (\overrightarrow{X_{int} - X_{goal}})}{\|X_{new} - X_{near}\| \cdot \|X_{int} - X_{goal}\|} \quad (3)$$

3.4 算法仿真实验验证

为验证改进RRT*算法,本文将使用Matlab仿真平台,在3个二维环境中每种算法分别进行50次实验,然后取平均值把改进RRT*和传统RRT*算法作对比。起点设置为[5,5],终点为[95,95],步长 ρ 为2,近邻节点选取半径为5。

3.4.1 算法在不同地图中仿真效果

传统RRT*及其改进算法在地图1中路径优化仿真结果如图7所示。从图7中可以看出,相比于传统RRT*算法(图7(a)),加入改进点1和2后

(图7(b)),路径更短、更平滑,并且随机树的分支趋向同一节点,加入改进点3后的RRT*算法(图7(c))避免了向多余空间探索,进一步提高了算法效率。传统RRT*及其改进算法在地图1中仿真结果见表1。从表1中可以看出,在扩展节点数大幅减少的情况下,平均时间减少了84.1%,路径长度减少了7.4%。

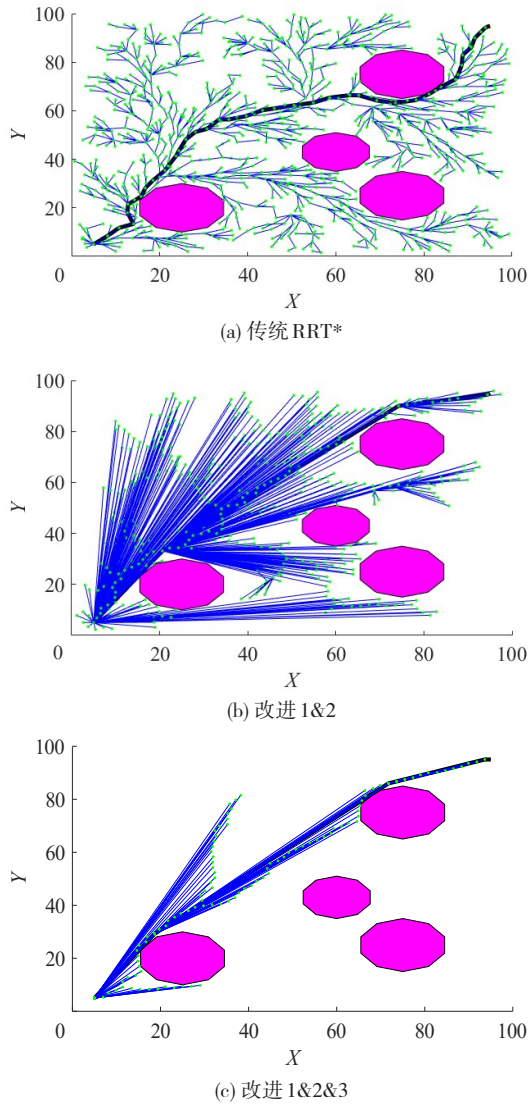


图7 传统RRT*及其改进算法在地图1中仿真效果

表1 地图1仿真结果

算法	耗时/s	路径长度	节点数
传统RRT*	0.63	141.86	1239
改进1&2	0.37	132.57	508
改进1&2&3	0.10	131.37	118

传统RRT*及其改进算法在地图2中路径优化仿真结果如图8和表2所示。在地图2中改进RRT*

算法每次运行平均时间减少了约72.2%,路径长度减少了约7.8%。

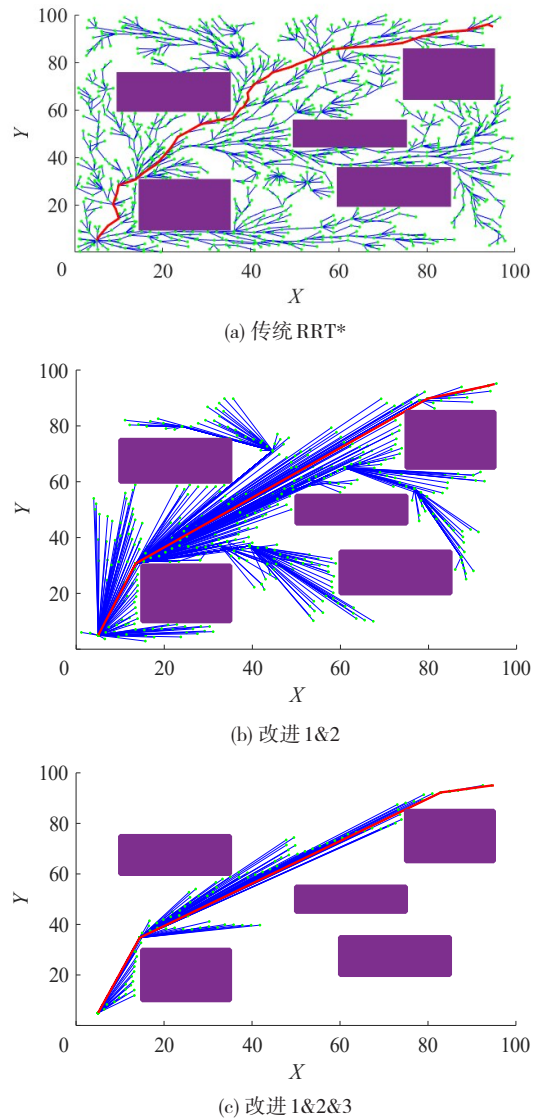


图8 传统RRT*及其改进算法在地图2中仿真效果

表2 地图2仿真结果

算法	耗时/s	路径长度	节点数
传统RRT*	0.18	143.49	1264
改进1&2	0.12	134.60	479
改进1&2&3	0.05	132.24	127

传统RRT*及其改进算法在地图3中路径优化仿真结果如图9和表3所示。在地图3这样的复杂环境中相比于传统RRT*算法(图9(a)),加入改进点3后的RRT*算法(图9(c))的扩展节点数大幅减少,每次运行的平均时间减少了81%,路径长度减少了约7.2%。

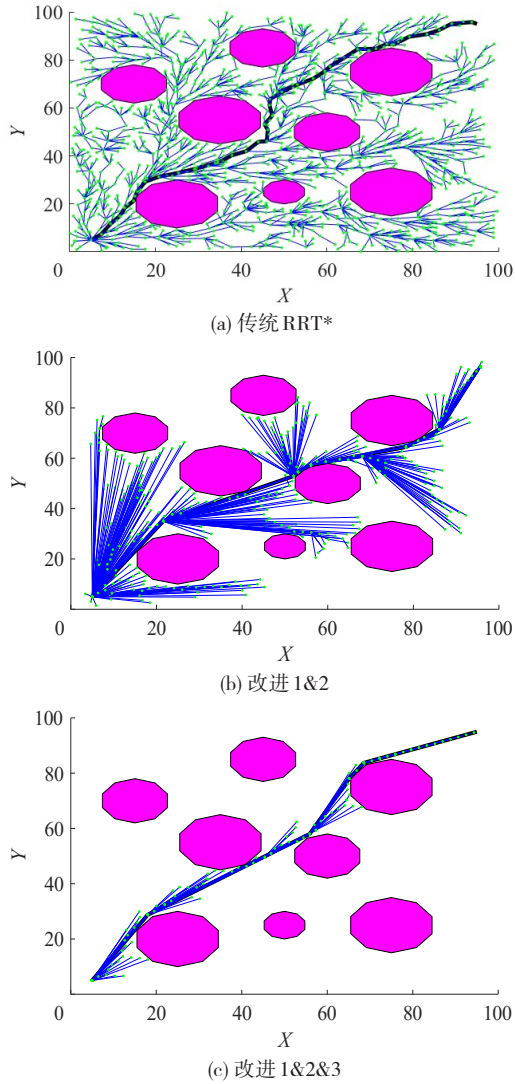


图9 传统 RRT*及其改进算法在地图3中仿真效果

表3 地图3仿真结果

算法	耗时/s	路径长度	节点数
传统 RRT*	1.42	143.84	1139
改进 1&2	0.81	134.08	454
改进 1&2&3	0.27	133.49	113

以上的仿真实验验证了改进方法的可行性,接下来将会把改进 RRT*算法、文献[5]中的目标引力 RRT*算法和文献[10]中的目标偏置 RRT*算法在地图3中分别进行50次仿真实验验证,然后对各个评价指标(时间、路径长度、扩展节点数、成功率)取平均值进行对比。

3.4.2 不同算法在地图中仿真对比

不同算法在地图中仿真对比将进行两轮实验,第一轮将偏置 RRT*算法和目标引力 RRT*算法的采样次数都设置为3000,将改进 RRT*算法的采样

次数设置为2500,仿真结果如图10和表4所示。

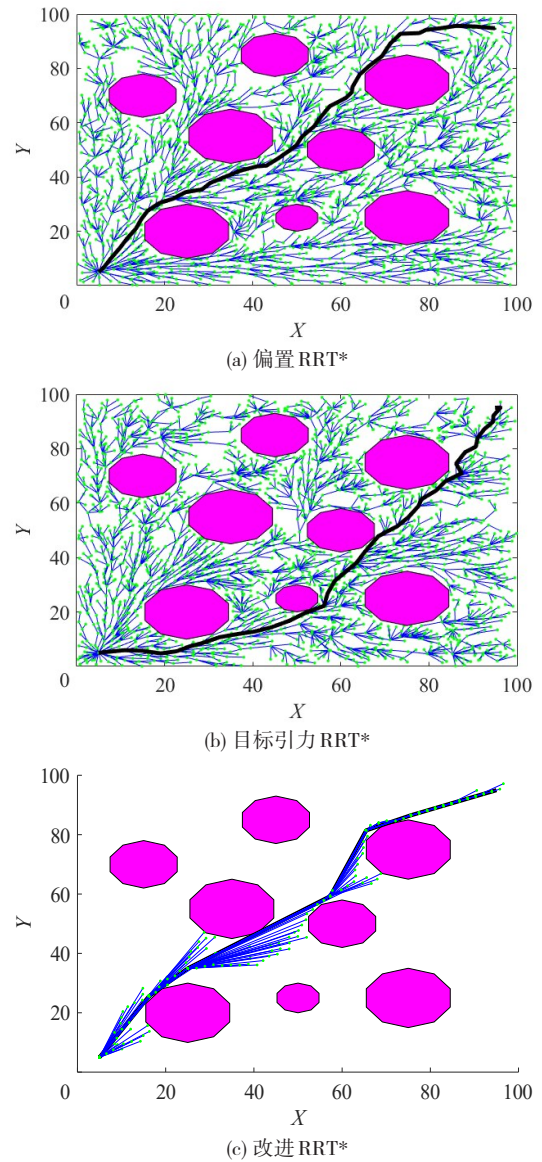


图10 第一轮对比实验

表4 第一轮对比实验结果

算法	耗时/s	路径长度	节点数	成功率/%
传统 RRT*	5.4	138.8	2389	100
目标引力 RRT*	4.5	140.0	2402	98
本文改进 RRT*	2.3	134.1	307	98

第二轮将偏置 RRT*算法和目标引力 RRT*算法的采样次数都设置为2000,将改进 RRT*算法的采样次数设置为1500,仿真结果如表5和图11所示。

表5 第二轮对比实验结果

算法	耗时/s	路径长度	节点数	成功率/%
传统 RRT*	6.9	140.2	1595	100
目标引力 RRT*	6.2	143.7	1604	86
本文改进 RRT*	1.6	133.2	176	76

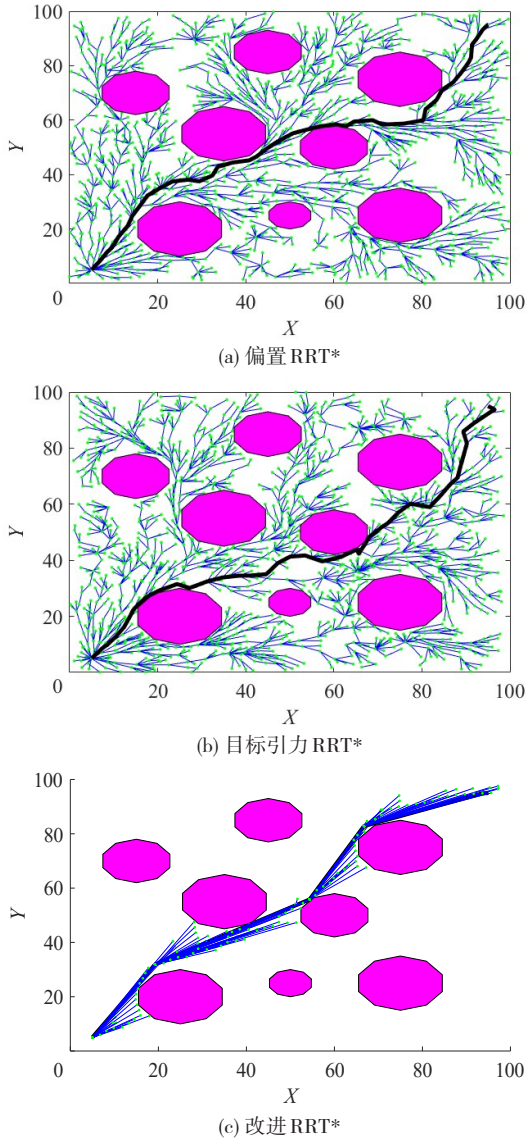


图 11 第二轮对比实验

两轮实验结果表明,尽管改进 RRT*在迭代次数较少的情况下成功率略低,但是相比于另外两种算法耗时更少,规划出来的路径质量更优,同时随机树节点占用的内存远小于另外两种算法。

4 仿真实验验证

将改进的 RRT*算法应用于机械臂的路径规划,采样点是笛卡尔坐标系中的三维矢量,如式(4)所示:

$$X_{rand} = [R_x, R_y, R_z] \quad (4)$$

设随机树中的每个节点为 9 维向量,如式(5)所示:

$$X_i = [p_x, p_y, p_z, \theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6] \quad (5)$$

其中, X_i 为随机树中的第 i 个节点, p_x, p_y, p_z 为路径规划算法得到的机械臂末端在三维空间中的一点 X_{new} 的坐标, $\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6$ 为相应的机械臂 6 个关节角度值,两者的关系如式(6)所示:

$$[\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6] = f(p_x, p_y, p_z, pitch, yaw, roll) \quad (6)$$

其中, f 为机械臂逆运动学求解函数,具体求解算法见文献[22], $pitch, yaw, roll$ 为机械臂末端执行器在整个路径规划过程中根据需求自行设定的相对于机械臂基座坐标系的抓取姿态参数, $pitch$ 为俯仰角(绕基坐标 y 轴的转角), yaw 为航向角(绕基坐标 x 轴的转角), $roll$ 为横滚角(绕基坐标 z 轴的转角)。由于在某些工业应用时,为了安全起见,需要机械臂保持一个稳定的抓取姿态去完成抓取任务,所以将 $roll$ 与 $pitch$ 设置为 0° , yaw 设置为 180° ,使得机械臂始终保持一个夹爪掌心朝下的抓取姿态,如图 12 所示。

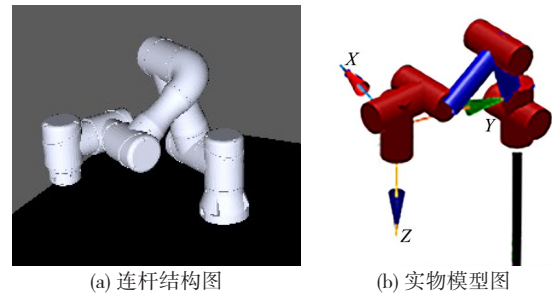


图 12 抓取姿态示例

现将改进 RRT*算法应用到机械臂路径规划中,改进的 RRT*算法步骤如下:

- 1) 确定起点、目标点位置和机械臂抓取姿态参数。
- 2) 采样一点 X_{rand} , 从随机树中找到距离 X_{rand} 最近的节点, 记作 X_{near} , 从节点 X_{near} 沿着 X_{near} 与 X_{rand} 的连线方向生成一点 X_{new} , 如图 13 所示。设置步长为 ρ , X_{new} 的计算公式如式(7)所示:

$$X_{new} = X_{near} + \frac{(X_{rand} - X_{near})}{\|X_{rand} - X_{near}\|} \cdot \rho \quad (7)$$

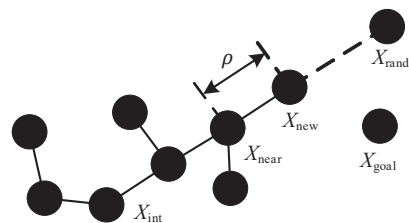


图 13 RRT*节点扩展图示例

3)在 X_{new} 处进行机械臂的逆运动学求解(见式(6)),得到各个关节角度。

4)通过正运动学求解得到各个关节在三维空间的位置,进而确定每个连杆的直线方程,然后检测每个连杆是否和障碍物相交^[23],如果相交则发生碰撞。

5)若没发生碰撞,则对 X_{new} 进行父节点重选,在将 X_{new} 加入随机树成为新的节点后,再进行剪枝。

6)若发生碰撞,则删除扩展节点 X_{new} ,并返回步骤2)。

7)如果达到最大迭代次数,则输出路径信息,否则返回步骤2)。

本文采用 Matlab 搭建仿真平台,其中的机械臂运动学模型由 Matlab 机器人工具箱搭建^[24],DH 参数见表6。

表6 DH参数

连杆 (i)	连杆转角 $\alpha_{i-1}/(^{\circ})$	杆长 a_{i-1}/mm	连杆偏距 d_i/mm	关节转角 $\theta_i/(^{\circ})$
1	0	0	144	$\theta_1(0)$
2	90	0	0	$\theta_2(-90)$
3	0	-264	0	$\theta_3(0)$
4	0	-236	106	$\theta_4(-90)$
5	90	0	114	$\theta_5(0)$
6	-90	0	67	$\theta_6(0)$

改进的 RRT* 迭代次数设置为 5000,球体和圆柱体为障碍物,起点设置为 $[0,400,200]$,目标点设置

为 $[300,10,450]$,机械臂的基座坐标设置为 $[0,0,0]$,步长设置为 15,近邻节点选取范围为 40。

改进 RRT* 算法的仿真结果如图 14,球体和柱体为障碍物,蓝色曲线为改进 RRT* 优化路径。

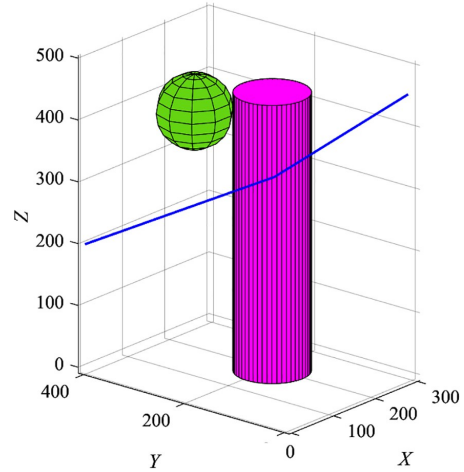


图 14 机械臂末端轨迹

将改进 RRT* 规划出来的关节角度数据进行线性插补 2000 次,得到数据 Q,然后在 Matlab 命令窗口输入指令“robot.plot(Q)”就能让机械臂模型沿着图中的蓝色轨迹进行运动,如图 15 所示。图 15(a)所示为机械臂的起点位置,图 15(b)–图 15(e)所示为机械臂运动过程的中间位置,图 15(f)所示为机械臂的目标点位置。机械臂关节角度变化如图 16 所示,从图中可以看到机械臂能有效避开障碍物到达目标点。

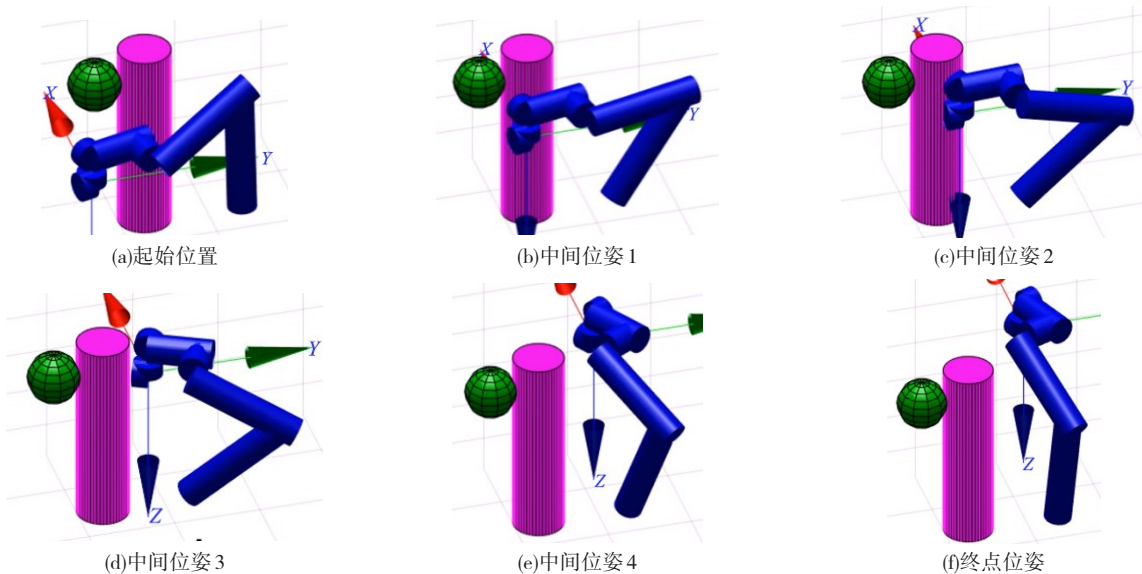


图 15 机械臂路径规划过程

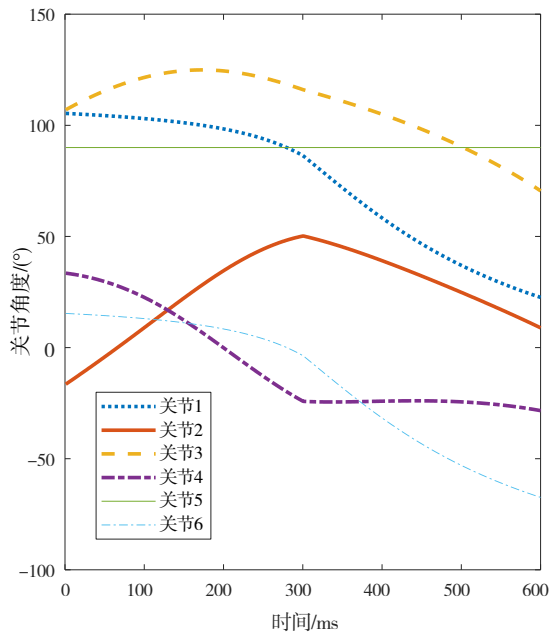


图16 关节角度变化曲线

综上所述,无论是路径规划时间、长度、还是占用内存,改进RRT*算法都要优于传统RRT*,能够在较短的时间内搜索出接近最优的最短期望路径。

参考文献:

- [1] 顾子侣,刘宇,岳广,等.基于改进RRT算法的快速路径规划[J].兵器装备工程学报,2022,43(10):294-299.
- [2] 李晓旭,马兴录,王先鹏.移动机器人路径规划算法综述[J].计算机测量与控制,2022,30(7):9-19.
- [3] 蔡文涛,邓屹,张静,等.基于改进RRT算法的机械臂路径规划[J].传感器与微系统,2019,38(5):121-124.
- [4] 孙立新,高建明,高春艳,等.改进RRT*算法的七自由度机械臂避障路径规划[J].控制工程,2022,18(22):1-8.
- [5] 马慧丽,鲁照权,王寿庭.基于改进RRT算法的机械臂路径规划研究[J].机械设计与研究,2020,36(4):42-46.
- [6] 张勤,乐晓亮,李彬,等.基于CTB-RRT*的果蔬采摘机械臂运动路径规划[J].农业机械学报,2021,52(10):129-136.
- [7] 张振,李新宇,董昊臻,等.基于约束采样RRT的机械臂运动规划方法[J].计算机集成制造系统,2022,33(13):1-18.
- [8] 双丰,刘旭元,李少东,等.基于GPF-RRT*的机械臂自主运动规划[J].计算机集成制造系统,2022,38(21):1-20.
- [9] QURESHI A H,AYAZ Y.Intelligent bidirectional rapidly-exploring random trees for optimal motion planning in complex cluttered environments [J].Robotics and Autonomous Systems,2015,68(1):1-11.
- [10] 李剑光,刘椿鹏,周梓燧,等.基于改进RRT*算法的机械臂路径规划[J].组合机床与自动化加工技术,2022,22(3):9-12.
- [11] 代伟,李创业,杨春雨,等.基于低差异序列与快速扩展随机树融合算法的机械臂路径规划[J].控制理论与应用,2022,39(7):130-144.
- [12] 王兆光,高宏力,宋兴国,等.基于GB_RRT算法的机械臂路径规划[J].机械设计与制造,2019,17(7):1-4.
- [13] 王雨,刘延俊,贾华,等.基于强化RRT算法的机械臂路径规划[J].山东大学学报(工学版),2022,23(25):1-9.
- [14] 王怀震,高明,王建华,等.基于改进RRT*-Connect算法的机械臂多场景运动规划[J].农业机械学报,2022,53(4):432-440.
- [15] 冯晨旭,胡三宝.基于改进天牛须搜索的机械臂运动学逆解[J].武汉大学学报(工学版),2020,53(12):1121-1128.
- [16] 袁杰,刘祚时.基于ROS的六自由度机械臂避障研究[J].制造业自动化,2017,39(12):41-43.

改进RRT*在三维空间中规划出较优路径的情况下,关节角度变化曲线整体较为平滑,没有突变(见图16),这表明改进RRT*算法可以很好地应用到机械臂路径规划中,能够让机械臂更加安全高效地工作。

5 结束语

本文采用了改进RRT*算法来解决工业机械臂在狭窄障碍物空间中运动规划的难题。改进算法改变了父节点重选的方式去优化路径;然后通过改进RRT*算法的采样方式,提高了算法收敛速度;为满足机器人运动学规律,引入节点拒绝策略,消除了路径中过大的转弯角。采用MATLAB进行了一系列仿真实验,证明了改进RRT*算法的可行性和有效性,可以快速地对机械臂进行路径规划,对机器人路径规划具有一定的参考价值。

- [17] 杜金财,尹建军,贺坤,等.葡萄采摘4-DOF机械臂设计与虚拟样机仿真[J].农机化研究,2019,41(8):58-64.
- [18] SAPRA P,KHALIL S,GUPTA S.Accounting for lack of trust in optional binary RRT models using a unified measure of privacy and efficiency[J]. Journal of Statistical Theory and Practice,2022,16(3):1-3.
- [19] 徐启航.基于改进型RRT-Connect算法的六自由度机械臂路径规划研究[D].青岛:青岛理工大学,2021.
- [20] 姚智龙,张小俊,王金刚.改进Bi-RRT*算法的自动泊车路径规划[J].机械科学与技术,2022,2(1):1-10.
- [21] 李伟,金世俊.基于人工势场法和启发式采样的最优路径收敛方法[J].计算机应用,2021,5(10):1-8.
- [22] 刘瀚泽.六自由度工业搬运机器人的路径规划算法研究[D].天津:天津理工大学,2021.
- [23] 李扬,张蕾,李鹏飞,等.基于改进RRT结合B样条的机械臂运动规划方法[J].计算机集成制造系统,2021(1):1-17.
- [24] 胡宇航,张尚盈,张晓艳.基于MATLAB的6R机械臂运动学的仿真研究[J].制造技术与机床,2022(11):27-32.

引用格式:

中文:刘学深,曹立佳.基于改进RRT*算法的机械臂路径规划研究[J].四川轻化工大学学报(自然科学版),2024,37(1):61-70.

英文:LIU X S,CAO L J.Research on path planning of manipulator based on improved RRT* algorithm[J].Journal of Sichuan University of Science & Engineering (Natural Science Edition),2024,37(1):61-70.

Research on Path Planning of Manipulator Based on Improved RRT* Algorithm

LIU Xueshen^{1a}, CAO Lijia^{1a,1b,2}

(1a.School of Automation and Information Engineering, 1b.School of Computer Science and Engineering, Sichuan University of Science & Engineering, Yibin 644000, China; 2. Key Laboratory of Higher Education of Sichuan Province for Enterprise Informationalization and Internet of Things, Yibin 644000, China)

Abstract: RRT (Rapidly expanding Random Tree) algorithm is a sampling-based path planning algorithm, which is quite suitable for robot path planning. However, the traditional RRT* algorithm has the disadvantages of long time consumption and high memory consumption utilization. Therefore, an improved RRT* algorithm is proposed to solve these problems, whose selection range of the parent nodes are optimized, and target biased sampling and heuristic strategy is introduced on the basis of the traditional random sampling mechanism, which reduces the time consumption of the algorithm and shortens the path length; the node rejection strategy is introduced to eliminate redundant paths with too large turning angles and further improve the algorithm efficiency. Matlab is used to carry out simulation experiments, and the results show that the improved RRT* algorithm can search the shortest collision free path from the starting point to the end point in a shorter time, and can be well applied to the path planning of the manipulator.

Key words: path planning; improved RRT*; target biased sampling strategy; heuristic strategy; node rejection strategy; mechanical arm