

基于改进 RRT 算法的六自由度机械臂路径规划

朱永浩, 李丹, 龚旭, 金仁才

(安徽工业大学 电气与信息工程学院, 安徽 马鞍山 243032)

摘要: 针对快速扩展随机树(RRT)算法路径代价大、采样过程慢的问题, 提出1种改进的RRT算法对六自由度机械臂进行路径规划。结合RRT*和RRT-connect算法的优点, 应用目标采样的思想加强算法向目标点搜索的趋向性, 引入贪婪思想提高算法效率, 结合五次B样条插值对路径进行平滑优化, 缩短规划路径的时间及长度; 利用Python中的matplotlib功能包统计RRT与改进的RRT算法规划所需时长、采样点数与路径长度, 且在ROS平台中利用动态规划库进行算法配置, 验证改进算法的路径规划效果。结果表明: 相比传统RRT算法, 采用所提改进算法对六自由度机械臂进行路径规划, 在平均路径长度上缩短了29.5%、在规划路径时间上缩短了8.5%, 路径规划成功率提高至96.7%, 验证了该算法在实际应用中的可行性。

关键词: 六自由度机械臂; 路径规划; RRT; 目标采样

中图分类号: TP 241.3 **文献标志码:** A **doi:** 10.12415/j.issn.1671-7872.23113



Path Planning of 6-DOF Manipulator Based on Improved RRT Algorithm

ZHU Yonghao, LI Dan, GONG Xu, JIN Rencai

(School of Electrical & Information Engineering, Anhui University of Technology, Maanshan 243032, China)

Abstract: An improved rapidly-expanding random tree (RRT) algorithm was proposed to plan the path of a six degree of freedom (DOF) robotic arm in response to the problem of high path cost and slow sampling process in the RRT algorithm. Combining the advantages of RRT * and RRT-connect algorithms, the target sampling was applied to enhance the tendency of the algorithm to search for target points, the greedy thinking was introduced to improve the efficiency of the algorithm, and the quintic B-spline interpolation was combined to optimize the path smoothness, shorten the path planning time and length. The matplotlib function pack in Python was used to calculate the time, number of sampling points, and path length required for RRT and improved algorithm planning, as well as the dynamic planning library for algorithm configuration on the ROS platform was used to verify the path planning effect of the improved algorithm. The results show that compared with the traditional RRT algorithm, the path planning of the six DOF robotic arm by the proposed improved algorithm shortens the average path length by 29.5%, the planning path time is shortened by 8.5%, and the success rate of path planning is increased to 96.7%, which verifies the feasibility of the algorithm in practice.

Keywords: six degree of freedom robotic arm; path planning; RRT; target sampling

随着机器人技术的发展, 机械臂具有工作空间大、操作简便、灵活性和自由度高等优点^[1], 在焊缝焊接、制造加工、拆卸装配、喷漆和码垛等领域得

到越来越广泛的应用^[2-4]。与此同时, 各项工作对机械臂的工作状态有更高的要求, 路径规划作为机械臂工作的关键技术, 受到越来越多的关注和研究。

收稿日期: 2023-07-27

基金项目: 安徽省自然科学基金项目(2108085MF225)

作者简介: 朱永浩(1996—), 男, 安徽滁州人, 硕士生, 主要研究方向为机械臂、自主导航等。

通信作者: 李丹(1976—), 女, 安徽马鞍山人, 博士, 教授, 主要研究方向为图像处理与机器人视觉、自主导航等。

引文格式: 朱永浩, 李丹, 龚旭, 等. 基于改进 RRT 算法的六自由度机械臂路径规划[J]. 安徽工业大学学报(自然科学版), 2024, 41(2):173-181.

机械臂的路径规划和智能小车类似,均是以机器人当前位置为起点,规划出 1 条运动到终点、无碰撞的最优路径^[5]。但是,机械臂的运动规划主要用于多维空间,规划难度更大。

近年,机械臂路径规划算法主要分为基于图搜索、基于势场、基于采样等类别^[6]。其中基于采样的算法无需预处理,也无需提前建图,在多维空间中得到更广泛的应用。快速扩展随机树 (rapidly-exploring random tress, RRT) 算法是机械臂路径规划中最常用的算法^[7],但该算法在复杂环境下会出现效率降低且无法得到最优路径的问题。基于 RRT 的改进算法, Kuffner 等^[8]提出让起点和终点同时生成随机树来进行双向拓展的 RRT-connect 算法; Ferguson 等^[9]提出动态快速搜索树 (dynamic-RRT, DRRT) 算法,通过给定目标拓展方向,可较大程度地降低规划时陷入局部最优的情况; Karaman^[10]提出改进的 RRT* 算法,通过添加图优化和剪枝理论,保留传统 RRT 算法的完备性,同时兼具渐进最优性,但大大增加了搜索时间; Islam 等^[11]提出 RRT*-smart 算法,利用智能采样使搜索路径快速收敛; Klemm 等^[12]结合 RRT* 的渐进最优和 RRT-connect 的双向搜索,提出使搜索路径朝理论最优解收敛的 RRT*-connect 算法; 朱宏辉等^[13]通过加入规避步长延伸法改进 RRT* 算法,但改进算法用于复杂路径的规划时,难以获得最优解; 陈肇星等^[14]提出的 DSSP-RRT 算法通过动态采样减少迭代次数,但存在可能陷入局部最优的情况。综上可看出:改进的 RRT 算法均在一定程度上优化了路径规划性能,但依然存在一些不足,且在高维空间下的机械臂路径规划通常是不稳定的,需在不同层面上做出改进。鉴于此,结合 RRT* 和 RRT-connect 算法的优点,提出 1 种改进的 RRT 算法对六自由度机械臂进行路径规划,并分别利用 Python 中的 matplotlib 功能包以及 ROS 平台中的动态规划库 (open motion planning library, OMPL),对睿尔曼 RM-65 机械臂进行避障路径规划仿真实验,结果显示该算法规划出的路径更优且用时更少。

1 机械臂运动学分析

1.1 运动学模型

以六自由度机械臂为对象,对睿尔曼 RM-65 机械臂进行正运动学分析。机械臂可视为由关节和连杆组成,求解机械臂末端相对于基准坐标系的位姿,即为机械臂的正运动学建模^[15]。机械臂结构

尺寸如图 1,其中 $d_1=240.5$ mm, $d_2=102.0$ mm, $d_3=256.0$ mm, $d_4=210.0$ mm, $d_5=144.0$ mm, $d_6=850.5$ mm, $d_7=81.5$ mm。

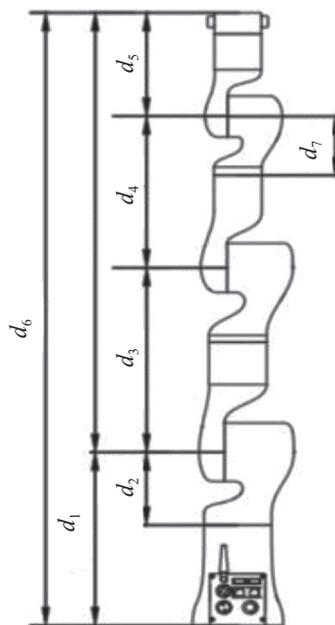


图1 RM-65 机械臂结构尺寸

Fig. 1 Structural dimensions of RM-65 robotic arm

基于改进的 MD-H 参数建模,建立的连杆坐标系如图 2。

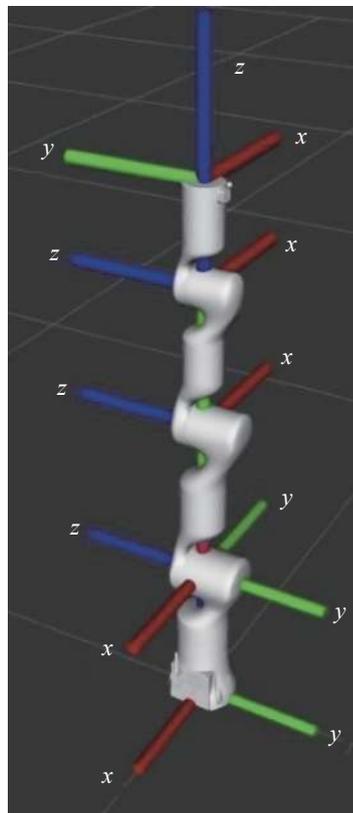


图2 连杆坐标系示意图

Fig. 2 Schematic diagram of the connecting rod coordinate system

x 轴、 y 轴、 z 轴如图 2 中标注所示, 机械臂最底部的坐标系 base_link 为机械臂的基坐标系, Link1~Link6 分别对应关节 1~6 的坐标系。根据机械臂的尺寸得到 MD-H 参数, 如表 1。

根据 MD-H 参数与各关节角度, 确定机械臂末端执行器基于机械臂基座的空间位姿, 矩阵 ${}^{i-1}T$ 为连杆 i 坐标系到连杆 $i-1$ 坐标系的 MD-H 变换矩阵, 该矩阵经以下四次运动变换得到, 具体变换公式如下:

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

其中: α_i 为连杆转角, 即当前坐标系 z 轴到下一坐标系 z 轴的角度; a_i 为连杆长度, 即当前坐标系 z 轴到下一坐标系 z 轴公垂线的长度; d_i 为连杆偏距, 即当前坐标系到下一坐标系需沿 z 轴平移的距离; θ_i 为关节角度, 即当前坐标系到下一坐标系需绕 z 轴旋转的角度。再根据 MD-H 参数表中的数据计算出相邻连杆间坐标系的变换矩阵:

$${}^0T_1 = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 240.5 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

$${}^1T_2 = \begin{bmatrix} -\sin\theta_2 & \cos\theta_2 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ \cos\theta_2 & -\sin\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

$${}^2T_3 = \begin{bmatrix} \sin\theta_3 & \cos\theta_3 & 0 & 256 \\ -\cos\theta_3 & \sin\theta_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

$${}^3T_4 = \begin{bmatrix} \cos\theta_4 & -\sin\theta_4 & 0 & 0 \\ 0 & 0 & 1 & 210 \\ -\sin\theta_4 & -\cos\theta_4 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

表 1 RM-65 机械臂 MD-H 参数

Tab. 1 RM-65 robotic arm MD-H parameter

连杆	扭角/(°)	长度/mm	偏距/mm	关节角度/(°)	运动范围/(°)
1	0	0	240.5	0	-178~178
2	90	0	0	90	-130~130
3	0	256	0	-90	-135~135
4	-90	0	210.0	0	-178~178
5	90	0	0	0	-128~128
6	-90	0	144.0	0	-360~360

$${}^4T_5 = \begin{bmatrix} \cos\theta_5 & -\sin\theta_5 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ \sin\theta_5 & \cos\theta_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

$${}^5T_6 = \begin{bmatrix} \cos\theta_6 & -\sin\theta_6 & 0 & 0 \\ 0 & 0 & 1 & 144 \\ -\sin\theta_6 & -\cos\theta_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

由式 (2)~(7) 可得出机械臂末端坐标系相对于底端基坐标系的运动学位姿变换矩阵, 即正运动学方程 0T_6 可表示为:

$${}^0T_6 = {}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4 {}^4T_5 {}^5T_6 \quad (8)$$

$${}^0T_6 = \begin{bmatrix} \vec{n} & \vec{o} & \vec{a} & \vec{p} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9)$$

其中: n, o, a 为机械臂末端互相垂直的 3 个坐标轴。 a 轴为接近轴 (approach axis), 在机械臂运动时一般以 a 轴去接近目标; o, n 分别为方向轴 (orientation axis) 和垂直轴 (normal axis); $\vec{n}, \vec{o}, \vec{a}$ 表示机械臂末端姿态; \vec{p} 表示机械臂末端的空间位置。为验证所建正运动学方程的正确性, 采用 Matlab2018a 的 Robotic Toolbox 工具箱, 测试仿真情况下机械臂末端实际位姿是否与使用 MD-H 参数法建模计算得

到的位姿一致,使用表1所示的MD-H参数构建机械臂,结果如图3。

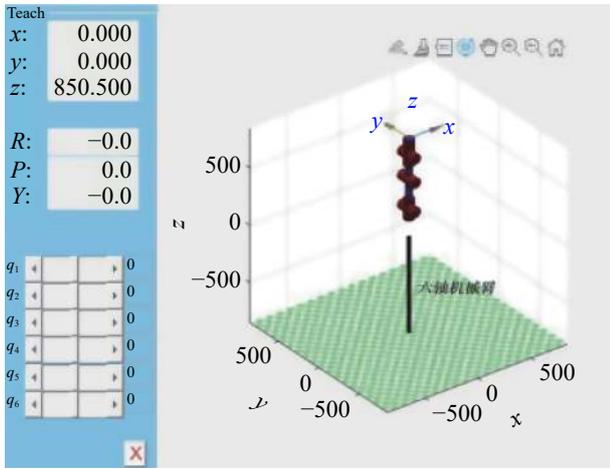


图3 Matlab 仿真结果示意图

Fig. 3 Schematic diagram of Matlab simulation results

图中:RPY为矩阵。 0T 中的旋转矩阵转化得到的欧拉角, $q_1 \sim q_6$ 为对应6个关节角度;取3组关节角度,分别通过Robotic Toolbox工具箱读取和建模得到正运动学方程计算,结果如表2。由表2可看出:Robotic Toolbox工具箱读取所得位姿与运动学方程计算所得位姿结果基本一致,表明由MD-H参数法建模计算得到的正运动学方程正确。

逆运动学建模是通过已知的机械臂末端位姿来确定满足期望位姿的机械臂各关节的状态^[16],常用的解法有数值求解法和解析求解法。其中数值求解法采用非线性优化求解,适用范围较广但计算量较大、速度较慢;解析求解法速度快但往往会得出多组奇异解。因此,文中选择数值求解法计算得到若干组逆解,根据关节角度范围和碰撞关系的检测在逆解中找出最优解。

表2 位姿对比结果

Tab. 2 Pose comparison results

关节角度/(°)	Matlab仿真所得位姿	运动学方程计算所得位姿
(0, 0, 0, 0, 0, 0)	$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 850.500 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 850.500 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
(90,30,30,0,60,0)	$\begin{bmatrix} 0 & -1 & 0 & 0 \\ -0.500 & 0 & -0.866 & -434.573 \\ 0.866 & 0 & -0.500 & 495.203 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 & -1 & 0 & 0 \\ -0.500 & 0 & -0.870 & -434.583 \\ 0.862 & 0 & -0.500 & 495.198 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
(-45,0,-45,45,-45,45)	$\begin{bmatrix} 0.500 & -0.500 & 0.707 & 206.800 \\ -0.500 & 0.500 & 0.707 & -3.177 \\ -0.707 & -0.707 & 0 & 645 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0.500 & -0.500 & 0.707 & 206.800 \\ -0.500 & 0.500 & 0.707 & -3.177 \\ -0.707 & -0.707 & 0 & 645 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

1.2 碰撞检测

机械臂各连杆的形状不规则,在进行碰撞检测时常使用包围盒法,利用简单几何体将机械臂包围起来。常用的包围盒法有轴对齐边界框包围盒(axis-aligned bounding box, AABB)、包围球、方向包围盒(oriented bounding box, OBB)等^[17-18]。AABB法采用平行于坐标轴的最小六面体包围对象,包围球法采用半径最小的球体包围对象,OBB法采用最小长方体包围对象。李智^[19]提出了子弹形的包围盒(bullet bounding box, BBB),包围盒模型为圆柱体搭配半球体的组合。使用该方法对机械臂进行包围时,对于旋转不会影响相邻连杆模型的轴体,采用圆柱体来包围;对于旋转会影响相邻连杆的轴体,采用半球来包围,这样可使包围盒更紧凑,且不会大量增加后续计算量。

文中采用BBB方法和包围球方法对机械臂包围盒进行包围,将机械臂与空间障碍物的碰撞检测简化为求解圆柱中心线与球体半径的关系问题以及圆柱中心线与AABB包围盒各平面的距离问题^[20]。根据机械臂连杆旋转是否会影响相邻连杆,对第一轴、第四轴和第六轴采用圆柱包围,对第二轴、第三轴和第五轴采用半球和球形包围。具体为:将第一轴、第一连杆和第二轴转换为子弹形包围,第二连杆和第三轴转换为球形包围,第三连杆、第四轴、第四连杆和第五轴转换形成子弹包围,第五连杆、第六轴和第六连杆视作1个圆柱体,末端执行器视为1个球体。使用包围盒法可对机械臂包围进行简化,根据简化的机械臂模型,对各包围体间距离进行求解,各包围体之间的最短距离可作为判断是否发生碰撞的指标,距离大于0则未发生碰撞,否则判断为发生碰撞。

2 RRT 算法的改进

RRT 算法是 1 种基于随机采样的快速路径规划算法,通过采样在工作空间全局搜索,以给定的起点为根节点,利用随机取点生成新的子节点使扩展树不断生长,具有出色的空间搜索能力,对高维空间的路径规划也具有出色的性能^[21]。RRT 算法的主要步骤如图 4,包括采样、寻找邻近节点、拓展根节点、生成候选节点、碰撞检测以及生成邻近节点。具体过程为:确定起点 (S) 与终点 (F),并明确起点与终点之间的障碍物信息;以起点为根节点,随机扩展采样,遍历树中所有节点,生成候选节点 T_n ,并以固定步长生成 1 个邻近节点 T_{n+1} ,若 T_n 与 T_{n+1} 之间不存在障碍物,则 T_n 及 T_{n+1} 加入扩展树,否则放弃此次采样点并重新采样;循环上一步骤,直到采样得到 1 个节点 T_{n+1} 为终点 F 或该节点距离终点 F 的距离小于设定的阈值,则规划路径成功并退出采样。

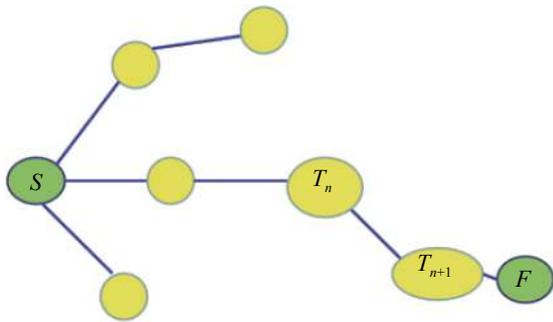


图4 RRT 算法示意图

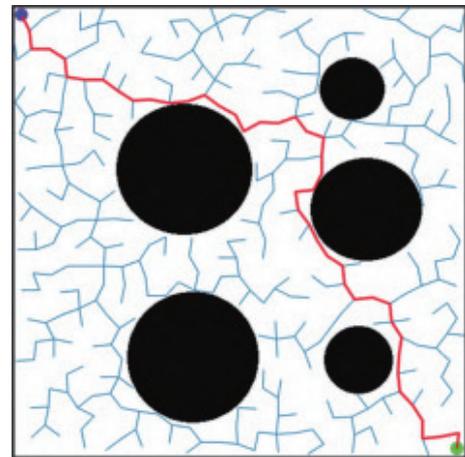
Fig. 4 Schematic diagram of RRT algorithm

RRT 算法在多维空间中适用性很强,但算法本身在搜索路径的过程中具有随机性,导致每次规划出的路径各不相同,规划出的是可行路径,并不是最优路径,且在复杂狭小环境中难以搜索到路径。为此,文中结合 RRT*算法和 RRT-connect 算法的优点,在搜索过程中,应用目标采样思想对路径进行调整与重新布线,使每次采样趋近给定的扩展方向,改进 RRT 算法。

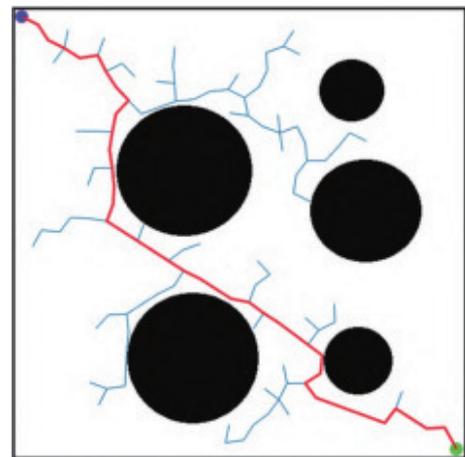
2.1 目标采样

RRT 算法在随机采样的过程中,搜索树时常会扩展到离目标很远的地方,即“无用区域”。为提高算法的效率,应用目标采样的思想加强算法向目标点搜索的趋向性,减少无用搜索,即人为引导采样过程中随机点的生成,在生成 1 个随机节点 T_n 时,以一定概率将节点生成在终点处,即 $T_n=T_F$ 。在提高算法效率的前提下,为保持搜索树对于未知空间的搜索能力,一般将概率 p 设定在 0.05~0.10 之间。具体步骤为:生成 1 个 (0,1) 的随机数 rand ,若 $\text{rand} \leq p$,

则节点生成在终点处,即 $T_n=T_F$;若 $\text{rand} > p$,则新节点保持随机生成,图 5 为加入目标采样前后的路径规划效果。



(a) 传统RRT算法路径规划



(b) 加入目标采样后的路径规划

图5 加入目标采样前后的路径规划效果

Fig. 5 Path planning effect before and after adding target sampling

从图 5 可看出:加入目标采样后的算法既能保证搜索树扩展时的随机性,又能一定程度上减免对“无用区域”搜索耗费的时间,在障碍物较少的空间效果较好。

2.2 贪婪思想

当随机采样到目标点附近区域时,受固定步长的限制,即使采样点与目标点之间无障碍物,搜索树也无法直接到达目标点,转而会继续进行无效搜索,使路径复杂、规划时长增加。针对这些采样点,引用贪婪思想,当搜索树生成的新节点与目标点间的距离在设定的阈值内时,判断新节点 $T_{n1} \sim T_{n5}$ 与目标点 F 直接相连的可行性,如图 6。当新节点与目标点之间存在障碍物时(如 T_{n1} , T_{n3} , T_{n4} , T_{n5}),该节点继续随机采样拓展;若之间不存在障碍物时(如 T_{n2}),

则该节点停止随机采样,且不受固定步长的限制,将该节点 T_{n2} 与目标点 F 直接相连,并将两点间的路径放入最终规划出的路径中。

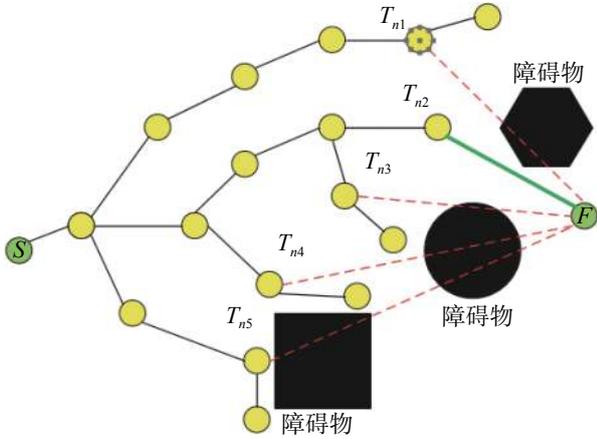


图6 引入贪婪思想的示意图

Fig. 6 Schematic diagram of introducing greedy thinking

2.3 路径平滑处理

在机械臂避障路径规划过程中,易在路径拐点出现急加速的情况,使规划过程的平稳性降低,故需对路径进行平滑处理。对于改进的 RRT 算法,选取路径中的若干拐点,使用 B 样条函数对拐点间的路径进行平滑处理。考虑到 B 样条次数越高,平滑处理后的路径越平滑,为保证最终规划出的路径效果,文中选择五次 B 样条函数进行平滑优化。根据需平滑优化的拐点坐标,构造出 1 条参数化的 B 样条曲线,使用取值范围为 $[0, 1]$ 的参数 t 表示曲线上的点;将参数 t 映射到实际的自变量范围,根据拐点个数将自变量范围划分成若干区间,并对每个区间使用五次 B 样条插值;根据插值结果可得到各区间的插值函数。位置表达式为

$$z(t) = k_0 + k_1(t-t_0) + k_2(t-t_0)^2 + k_3(t-t_0)^3 + k_4(t-t_0)^4 + k_5(t-t_0)^5 \quad (10)$$

求一阶导数得到速度表达式为

$$\frac{dz(t)}{dt} = k_1 + 2k_2(t-t_0) + 3k_3(t-t_0)^2 + 4k_4(t-t_0)^3 + 5k_5(t-t_0)^4 \quad (11)$$

求二阶导数得到加速度表达式为

$$\frac{d^2z(t)}{dt^2} = 2k_2 + 6k_3(t-t_0) + 12k_4(t-t_0)^2 + 20k_5(t-t_0)^3 \quad (12)$$

再根据给定的起始位置 z_0 和终点位置 z_1 、规划时间 T_i , 解出位置差 h 、速度 v_0 和 v_1 、加速度 c_0 和 c_1 以及五次多项式中的 6 个系数,满足的约束方程表达式为:

$$z(t_0) = z_0, z(t_1) = z_1, \frac{dz(t_0)}{dt} = v_0, \frac{dz(t_1)}{dt} = v_1, \frac{d^2z(t_0)}{dt^2} = c_0, \frac{d^2z(t_1)}{dt^2} = c_1 \quad (13)$$

其中,将式(13)代入式(10)~(12),解得:

$$T_i = t_1 - t_0, h = z_1 - z_0, k_0 = z_0, k_1 = v_0, k_2 = \frac{c_0}{2},$$

$$k_3 = \frac{1}{2T_i^3} [20h - (8v_1 + 12v_0)T_i - (3c_0 - c_1)T_i^2],$$

$$k_4 = \frac{1}{2T_i^4} [-30h + (14v_1 + 16v_0)T_i - (3c_0 - 2c_1)T_i^2],$$

$$k_5 = \frac{1}{2T_i^5} [12h - 6(v_1 + v_0)T_i + (c_1 - c_0)T_i^2] \quad (14)$$

2.4 改进 RRT 算法的实现

改进的 RRT 算法流程图如图 7。

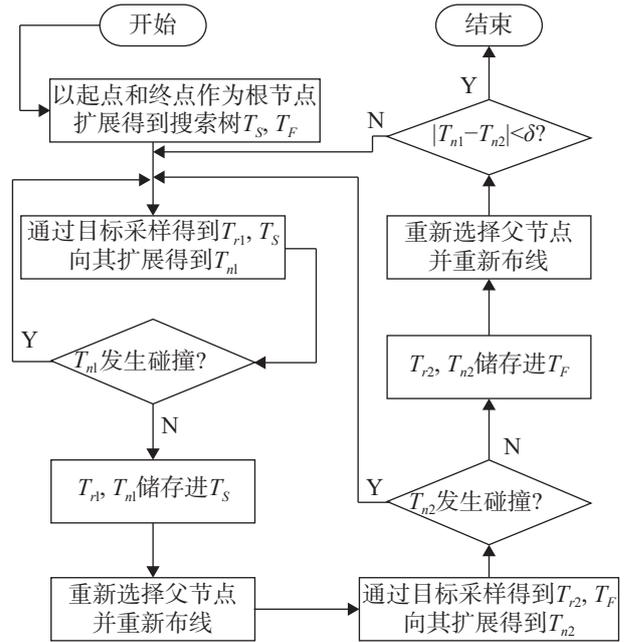


图7 改进的 RRT 算法流程图

Fig. 7 Flowchart of improved RRT algorithm

由图 7 可看出:采用改进的 RRT 算法进行路径规划,其主要步骤如下:

1) 确定起点与终点,将起点和终点分别作为根节点向空间中扩展生成搜索树,2 棵树分别为 T_S , T_F 。

2) 加入目标采样,使搜索树 T_S 按照给定方向扩展,采样得到 1 个新节点 T_{r1} ,检查搜索树 T_S 与 T_{r1} 最近节点 T_{n1} 之间是否存在障碍物。若不存在障碍物,则将节点 T_{n1} 储存进搜索树;若存在障碍物则重复步骤 2),直到找到 1 个无碰撞的节点 T_{n1} 。

3) 以 T_{n1} 附近规定的半径范围内寻找路径代价更小的邻近节点作为父节点,并重新布线,调整经过新节点 T_{n1} 的路径。

4) 与步骤 2) 类似,使搜索树 T_F 按照给定方向扩展,通过碰撞检测,采样得到 1 个无碰撞的节点 T_{n2} 。

5) 以 T_{n2} 附近规定的半径范围内寻找路径代价更小的邻近节点作为父节点,并重新布线,调整经过节点 T_{n2} 的路径。

6) 循环步骤 2) 到 5),直到采样得到的节点 T_{n1} 与 T_{n2} 重合或 2 个节点间的距离小于设定的阈值,则规

划路径成功并退出采样, 得到的 T_s 至 T_{n_1} 的路径、 T_F 至 T_{n_2} 的路径加上 T_{n_1} 至 T_{n_2} 的路径即为最终规划出的路径。

3 仿真实验与结果分析

为验证改进 RRT 算法的路径规划效果, 利用 Python 中的 matplotlib 功能包以及 ROS 平台的动态规划库 (open motion planning, OMPL), 对改进 RRT 算法与传统 RRT 算法进行路径规划仿真实验。

3.1 三维空间仿真实验与结果分析

三维空间地图范围单位大小为 $100 \times 100 \times 100$, 设置路径规划的起点坐标为 (10,10,10), 终点坐标为 (95,95,95), 如图 8。图中冗余线段为拓展树的树结构, 从起点连接到终点的线段为传统 RRT 算法和改进 RRT 算法最终规划出的路线。地图中的障碍物为 8 块正方体, 初始设定采样长度为 5、步长为 5, 当采样次数超过 1 000 次时认定此次路径规划失败, 放弃采样。

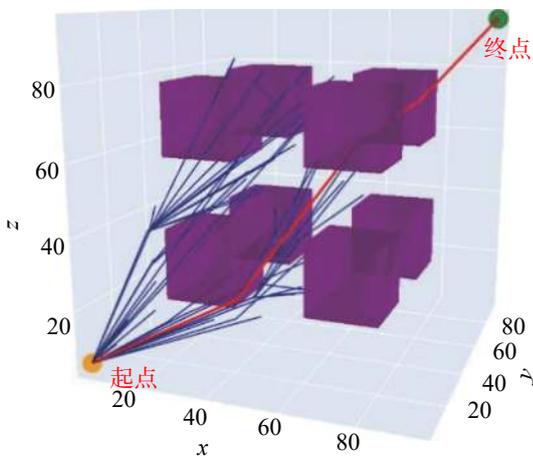
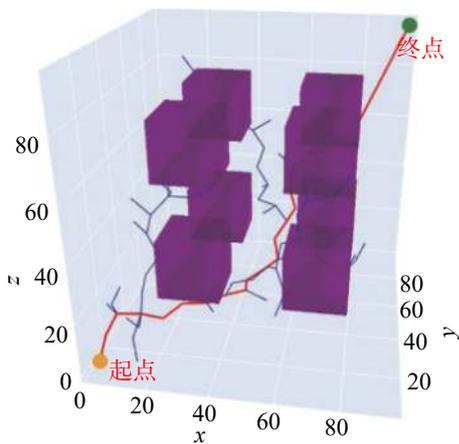


图8 三维空间仿真实验示意图

Fig. 8 Schematic diagram of three-dimensional space simulation experiment

在相同仿真环境下, 分别使用传统和改进的 RRT 算法进行 50 次路径规划实验, 对比 2 种算法规划的平均路径长度、计算时长和采样点数, 结果如表 3。

表 3 2 种算法的路径规划实验结果

Tab. 3 Experimental results of path planning for two algorithms

算法	单位平均 路径长度	平均计算 时长/s	平均采样 点数
传统RRT	292.6	1.264	128.2
改进RRT	206.2	1.156	141.5

由表 3 可看出: 与传统 RRT 算法相比, 改进 RRT 算法平均采样点数略高, 但规划出的单位平均路径长度从 292.6 减少至 206.2, 即规划出的路径代价大幅减少, 且平均计算时长从 1.264 s 减至 1.156 s, 改进 RRT 算法规划出的路径更优且用时更少。

3.2 ROS 仿真实验与结果分析

为验证改进的 RRT 算法在加入真实机械臂后的路径规划效果, 使用六自由度睿尔曼 RM-65 机械臂在 ROS 平台下进行仿真实验, 操作系统为 Ubuntu18.04。使用 Moveit! Setup Assistant 对机械臂进行配置, 具体流程如图 9。

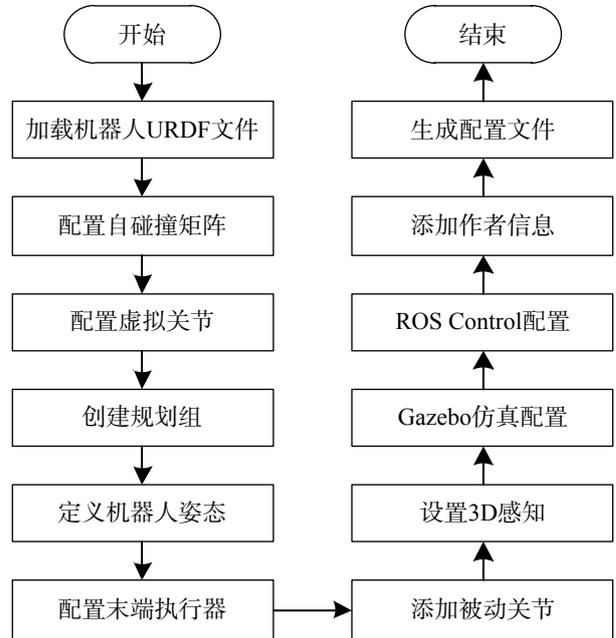


图9 Moveit! Setup Assistant 配置流程图

Fig. 9 Flowchart of Moveit! Setup Assistant configuration

在 ROS 平台下的动态规划库 (open motion planning library, OMPL) 中进行算法匹配, 配置完成利用 Moveit! 及其插件接口实现机械臂在仿真环境下的避障运动规划。如图 10 所示绿色机械臂为机械臂起始位姿, 橙色机械臂为目标位姿, 绿色长方体

为障碍物,机械臂从起始位姿开始经过路径规划并避开障碍物最终到达目标位姿视为实验成功。

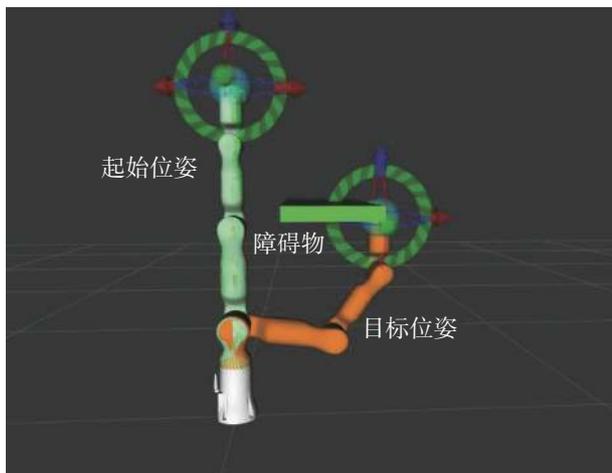
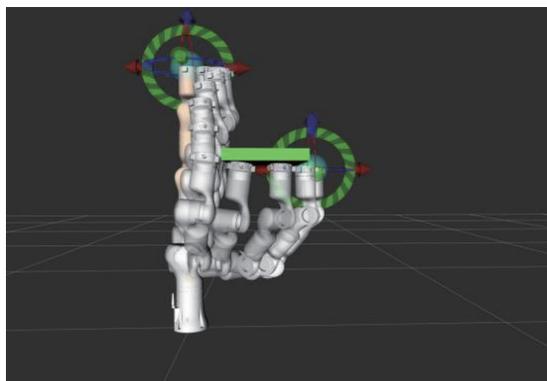


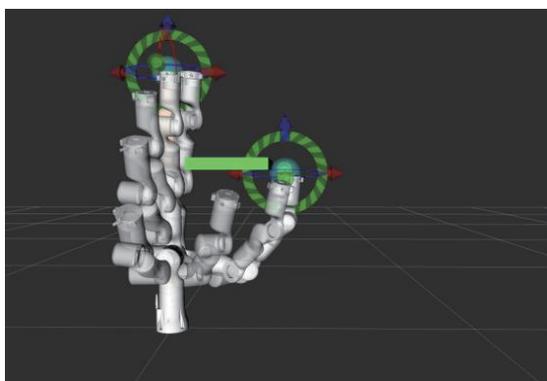
图10 避障运动规划示意图

Fig. 10 Schematic diagram of obstacle avoidance movement planning

使用传统 RRT 算法和改进的 RRT 算法分别进行 30 次实验,2 种算法的运动规划过程如图 11。在选用轨迹可视化功能后,图中连续出现的机械臂即为运动规划过程中的路径轨迹,机械臂轨迹越多说明规划时间越长、路径越复杂。



(a) 传统RRT算法



(b) 改进的RRT算法

图11 ROS 平台下避障运动规划示意图

Fig. 11 Schematic diagram of obstacle avoidance motion planning on ROS platform

从图 11 可看出:在相同实验环境和运动参数下,相比于传统 RRT 算法,改进的 RRT 算法规划时间和更短、规划的路径更优,且路径规划成功率提高了 10 个百分点,2 种算法平均规划时间和规划成功率统计如表 4。

表 4 避障运动规划实验结果

Tab. 4 Experimental results of obstacle avoidance movement planning

算法	平均规划时长/s	成功率/%
传统RRT	4.62	86.7
改进RRT	3.78	96.7

4 结论

提出 1 种改进的 RRT 算法对六自由度机械臂进行路径规划,结合 RRT*和 RRT-connect 算法的优点,应用目标采样的思想加强算法向目标点搜索的趋向性,引入贪婪思想提高算法效率,结合五次 B 样条插值对路径平滑优化,使机械臂可在较短时间规划出更优路径。利用 Python 中的 matplotlib 功能包统计传统与改进 RRT 算法规划路径所需的时间、采样点数量与路径长度,且在 ROS 平台中利用动态规划库对所提改进与传统 RRT 算法进行仿真实验。结果表明:采用所提改进算法对六自由度机械臂进行路径规划,可规划出路径代价更小的路径且用时更少,最终路径也更为平滑,验证了该算法在实际机械臂避障路径规划中的可行性。

参考文献:

- [1] KIM S H, NAM E, HA T I, et al. Robotic machining: a review of recent progress[J]. *International Journal of Precision Engineering and Manufacturing*, 2019, 20(9):1629-1642.
- [2] LIM G H, LAU N, PEDROSA E, et al. Precise and efficient pose estimation of stacked objects for mobile manipulation in industrial robotics challenges[J]. *Advanced Robotics*, 2019, 33(13):636-646.
- [3] RODRIGUEZ G D, SORROSAL G, CABANES I, et al. Human-robot interaction review: challenges and solutions for modern industrial environments[J]. *IEEE Access*, 2021, 9:108557-108578.
- [4] FANG H C, ONG S K, NEE A Y C. Robot path planning optimization for welding complex joints[J]. *The International Journal of Advanced Manufacturing Technology*, 2017, 90(9):3829-3839.
- [5] WANG X, LUO X, HAN B, et al. Collision-free path

- planning method for robots based on an improved rapidly-exploring random tree algorithm[J]. *Applied Science*, 2020, 10(14):1381–1385.
- [6] 彭求志, 田丽, 吴道华, 等. 基于混合算法的机器人路径规划[J]. *安徽大学学报(自然科学版)*, 2021, 45(5):77–83.
- [7] 周益邦, 章兰珠, 徐海铭. 基于改进双向RRT*算法的机械臂路径规划[J]. *计算机应用*, 2022, 42(S1):342–346.
- [8] KUFFNER J J, LAVALLE S M. 基于改进的RRT*-connect算法机械臂路径规划[J]. *计算机工程与应用*, 2021, 57(6):274–278.
- [9] FERGUSON D, KALRA N, STENTZ A. Replanning with RRTs[C]// 2006 IEEE International Conference on Robotics and Automation. Orlando, Florida: IEEE, 2006:1243–1248.
- [10] KARAMAN S, FRAZZOLI E. Sampling-based algorithms for optimal motion planning[J]. *The International Journal of Robotics Research*, 2011, 30(7):846–894.
- [11] ISLAM F, NASIR J, MALIK U, et al. RRT*-smart: rapid convergence implementation of RRT* towards optimal solution[C]//2012 IEEE International Conference on Mechatronics and Automation. Chendu: IEEE, 2012:1651–1656.
- [12] KLEMM S, OBERLANDER J, HERMANN A, et al. RRT*-connect: faster, asymptotically optimal motion planning[C]//2015 IEEE International Conference on Robotics and Biomimetics. Zhuhai: IEEE, 2015:1670–1677.
- [13] 朱宏辉, 明瑞冬, 朱轶. 基于改进RRT*算法的路径规划[J]. *中国机械工程*, 2018, 19(10):72–76.
- [14] 陈肇星, 徐向荣, 查文斌, 等. 基于改进RRT算法的7-DOF机械臂运动规划[J]. *制造业自动化*, 2022, 44(2):118–123.
- [15] 王文萍, 张培恒, 卫勇. 基于Kinova Mico六自由度机械臂运动学及其路径规划的研究[J]. *现代机械*, 2022(231):58–65.
- [16] JIN M, LIU Q, WANG B, et al. An efficient and accurate inverse kinematics for 7-DOF redundant manipulators based on a hybrid of analytical and numerical method[J]. *IEEE Access*, 2020, 8(5):16316–16330.
- [17] 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.
- [18] 杨帆. 基于B+树储存的AABB包围盒碰撞检测算法[J]. *计算机科学*, 2021, 48(S1):331–333, 348.
- [19] 李智. 冗余双臂机器人运动规划与操作的研究[D]. 天津: 天津理工大学, 2021.
- [20] ZHANG T, HONG J D. Collision detection method for industrial robot based on envelope-like lines[J]. *Industrial Robot: the International Journal of Robotics Research and Application*, 2019, 46(4):510–517.
- [21] 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, 52(6):123–130.

责任编辑: 何莉