

基于改进RRT算法的机械臂路径规划仿真

范彦清

上海理工大学机械工程学院, 上海

收稿日期: 2024年6月22日; 录用日期: 2024年7月16日; 发布日期: 2024年7月25日

摘要

快速随机扩展树(RRT)算法在机械臂路径规划过程中规划路径过程耗时长, 规划方式具有的随机性, 在实际的生产过程中严重影响路径规划的效率, 因此对于RRT算法的改进成为了路径规划算法的重要研究方向。文中采用高斯采样的策略对RRT算法进行改进, 提出了具有方向性搜索策略的改进RRT算法并应用于六自由度机械臂的路径规划上。通过MATLAB平台进行机械臂的避障仿真实验。通过实验结果表明, 改进RRT算法可以有效的节约搜索时间, 并提高路径规划的成功率, 也为机械臂路径规划的控制提供一些思路。

关键词

机械臂, RRT算法, 高斯采样, 路径规划

Simulation of Robotic Arm Path Planning Based on Improved RRT Algorithm

Yanqing Fan

School of Mechanical Engineering, University of Shanghai for Science and Technology, Shanghai

Received: Jun. 22nd, 2024; accepted: Jul. 16th, 2024; published: Jul. 25th, 2024

Abstract

Rapid Randomized Expanded Tree (RRT) algorithm in the robotic arm path planning process planning path process is time-consuming, the randomness of the planning method, in the actual production process seriously affects the efficiency of path planning, so the improvement of RRT algorithm has become an important research direction of path planning algorithms. In this paper, the Gaussian sampling strategy is used to improve the RRT algorithm, and the improved RRT algorithm with directional search strategy is proposed and applied to the path planning of six-degree-of-freedom robotic arm. The simulation experiment of obstacle avoidance of the robotic arm is

carried out by MATLAB platform. The experimental results show that the improved RRT algorithm can effectively save search time and improve the success rate of path planning, and also provide some ideas for the control of robotic arm path planning.

Keywords

Manipulator Arm, RRT Algorithm, Gaussian Sampling, Path Planning

Copyright © 2024 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

机械臂路径规划是机械臂运动控制中的一个核心问题，也是机器人研究的热点领域，为了保证机械臂能够更好地完成复杂环境下的任务，对于其路径规划的研究有着重要意义[1]。

针对机械臂的路径规划问题国内外都有着大量的研究。在解决一些二维平面空间的路径规划问题时，使用 A*[2]、遗传算法[3]。就能取得很好的效果，但机械臂通常具有多自由度要实现空间抓取，基于二维平面的算法并不能满足实际的需求。为了解决空间路径规划问题，La Valle [4]。提出了基于采样的快速搜索随机树(RRT)算法，但 RRT 基础算法搜索速度慢、生成路径拐点多、当环境复杂度变高时，RRT 算法的规划效率会大大降低甚至可能寻找不到可行路径。为了提高 RRT 算法在复杂环境下的规划成功率，衍生出许多基于 RRT 算法的变种。RRT-Connect 算法通过从起始节点和目标节点交替生长[5]。这样就可以保证准确地到达单个目标状态，而不只是目标的一个集合。而且在大多数情况下，因为起始节点和目标节点交替生成路径与 RRT 算法相比，具有更高的效率。

本文针对 RRT 算法路径规划速度慢、在复杂环境求解困难的特点，加入高斯采样降低算法的随机性，并加入路径惩罚函数还优化生成路径，最后在 MATLAB 上。

2. 机械臂模型建立及碰撞检测

2.1. 机械臂运动学建模

首先对机械臂利用 D-H 参数法进行建模，D-H 的 4 个参变量的含义为： θ_i 为关节角， α_i 为连杆偏转角， a_i 为连杆长度， d_i 为连杆偏移量。机械臂具体 D-H 参数见表 1。在机械臂各连杆关节轴处建立空间直角坐标系，其中关节 1 的坐标系与基坐标系重合，其各杆关系如图 1 所示。

Table 1. Manipulator D-H parameter table

表 1. 机械臂 D-H 参数表

连杆 i	θ_i (°)	α_i (°)	a_i (mm)	d_i (mm)
1	0	90	14	330
2	-90	0	265	0
3	0	90	30	0
4	0	-90	0	270
5	0	90	0	0
6	-90	0	0	70

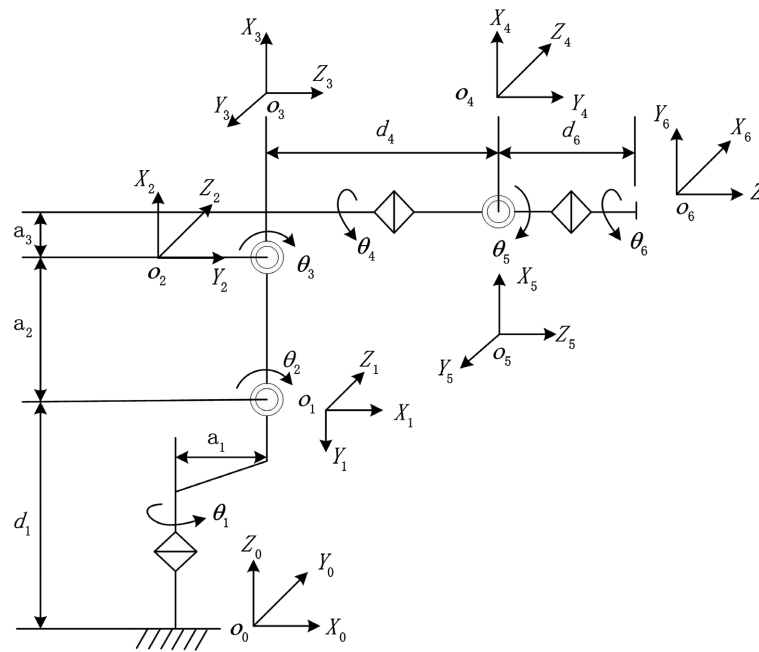


Figure 1. Manipulate link coordinate system
图 1. 机械臂连杆坐标系

在 D-H 参数法的基础上, 6 自由度机械臂相邻连杆坐标系之间的齐次变换矩阵为:

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

6 自由度机械臂从基座坐标系到末端抓取执行器的位姿变换矩阵 T_6^0 如式(2)所示。

$$T_6^0 = A_1^0 A_2^1 A_3^2 A_4^3 A_5^4 A_6^5 = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

式中, 左上三阶矩阵表示末端抓取执行器的姿态, $[p_x \ p_y \ p_z]^T$ 表示末端的空间位置。

2.2. 机械臂碰撞检测.

机械臂路径规划过程中还需要对障碍物进行碰撞检测, 为了简化障碍物对其进行一定适当的抽象化, 将障碍物抽象简化为长方体, 即采用长方体包络检测法对机械臂进行碰撞检测。

而对于机械臂的连杆则简化为规则的圆柱体。将长方体包络法处理的障碍物, 视情况分成连杆圆柱体与长方体障碍物各面异面、相交、平行。使用长方体包络法处理的障碍物的各面均与坐标轴垂直, 基于以上原因, 只需要判断在三个方向的轴线上是否发生重叠来判断碰撞的发生与否。而对于机械臂连杆之间的碰撞, 可以通过检测两圆柱体中心轴线之间的距离来进行判定。

3. 算法原理

3.1. RRT 算法及其衍生算法简介

RRT 算法的核心思想是在随机生成的节点中不断扩展树，直到找到目标点或无法找到解为止。在扩展过程中，RRT 算法会选择与现有树距离最近的节点进行扩展，以保持树的连通性和覆盖性。RRT 算法并不能保证找到全局最优解，因为它是基于随机生成的节点进行搜索的。此外，当环境复杂度较高或存在障碍物时，RRT 算法可能会遇到困难。为了提高 RRT 算法在复杂环境下的规划成功率，衍生出许多基于 RRT 算法的变种。RRT-Connect 算法通过从起始节点和目标节点交替生长，这样就可以保证准确地到达单个目标状态，而不只是目标的一个集合。而且在大多数情况下，因为起始节点和目标节点交替生成路径与 RRT 算法相比，具有更高的效率。但由于采样的方向具有随机性，RRT 和 RRT-Connect 算法的规划结果通常不会收敛到最优解。渐进最优的 RRT* 算法在原有 RRT 算法的基础上改进了父节点筛选机制，利用 A* 算法的惩罚函数将采样中惩罚值最小的节点作为父节点，从而保证规划渐进最优。为了进一步提高收敛到渐进最优的速度，Informed RRT* 算法应运而生，该算法在 RRT* 算法的基础上通过对采样空间中采样点的分布模型进行一系列的数学优化，先基于原始的 RRT* 算法形成一条初始路径，之后在此路径的起始点和目标点作为焦点形成一个椭圆形区域(若为空间区域则形成一个椭球)，之后的采样点只在此椭圆区域中集中生成，以不断尝试搜寻此区域内的最短路径。

3.2. RRT 算法的改进

3.2.1. 高斯分布采样

高斯分布采样是在数据聚类中普遍使用的采样方式，该方法通过限制空间采样区域，对 RRT 算法的搜索空间进行约束，随着迭代次数的增加，在障碍物密集或复杂环境会生成大量的采样点，而在无障碍区域只会生成少量的采样点。在采样点密集的区域 RRT 算法的随机采样策略被众多采样点所取代，进而加快路径规划的过程。高斯密度函数的 n 维函数表达式如式(3)所示。

$$\phi(c; \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{c^2}{2\sigma^2}} \quad (3)$$

式中 σ 为高斯函数的标准差。

定义障碍物 $obs(c)$ ，与障碍物发生碰撞时，其值为 1，未发生碰撞时为 0。以便针对性的进行障碍物区域进行采样。空间点被采样的概率如下式所示。

$$f(c; \sigma) = \int obs(y) \phi(c - y; \sigma) dy \quad (4)$$

可以从式(4)中可以看出，当采样点周围障碍物约密集，函数的值越大。为防止采样器在障碍物内部采样，定义式(5)，当采样点出现在障碍物中时式(5)值为 0。当在障碍物之外时，其结果仍等于式(4)。

$$g(c; \sigma) = \max(0, f(c; \sigma) - obs(c)) \quad (5)$$

3.2.2. 冗余采样点的处理

在进行高斯分布采样策略后的 RRT 算法有效提高了搜索的效率，但仍旧有许多冗余点，会使得 RRT 算法生成的路径大都像锯齿状，不利于最优路径的产生，为了克服这个问题，引入最小路径代价函数。

在第一阶段，先使用高斯采样找到 x_{new} ，然后对其进行碰撞检测，没有发生碰撞的情况下，以 x_{new} 为圆心， r 为半径生成一个圆，将 x_{new} 暂定为 x_{min} ，那么从初始点到 $x_{nearest}$ 和 x_{min} 到 x_{new} 的路径暂定为最小路径代价 c_{min} ，接下来循环遍历附近节点集合内所有的点，如果新连接的路径花费小于 c_{min} 并且新连接的点

碰撞检测合格, 那么将该新的 x_{near} 作为最小的 x_{min} , 最小路径代价则是该节点到初始节点和该节点到 x_{new} 的总和, 最后将连接的边添加到原始路径之上。

在第二阶段, 重新连接原始路径的边, 以得到最优的路径, 遍历附近节点剩下的其他高斯采样点, 如果存在某一 x_{near} 到 x_{new} 的路径代价加上初始点到 x_{new} 的路径代价小于该 x_{near} , 且满足碰撞检测的要求情况下, 可将 x_{near} 暂定为父节点, 然后删除一切与父节点相连的其他所有的 x_{near} , 并且重新连接那条路径。具体过程如图 2 所示。

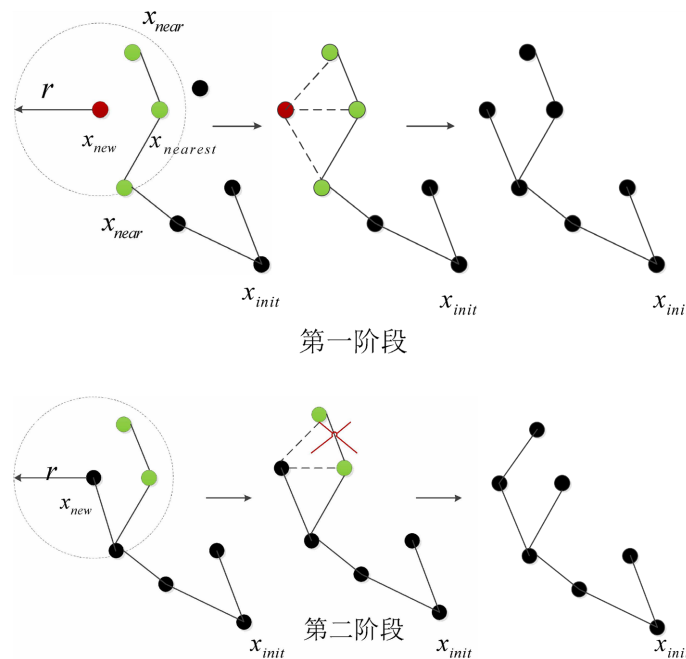


Figure 2. Minimum path cost function update process
图 2. 最小路径代价函数更新过程

其中路径代价函数是从起始点到该点的路径代价。相对于原算法加入路径代价函数可以有效地使规划的更为平滑, 减少了不必要的路径弯折, 很大程度上提高了机械臂运行的效率。

4. 仿真实验

为了验证改进的 RRT 算法的有效性, 在 MATLAB 中对静态障碍物分别进行二维和三维的实验仿真, 并对其两种算法进行比较, 在二维环境中, 将障碍物设置为一系列不规则的图形, 三维环境中设置成球形, 此外考虑到算法存在一定的随机性, 针对不同的实验场景, 分别进行 80 次模拟仿真实验, 将二维环境步长设置为 20, 三维环境步长设置为 10。通过 80 次模拟实验可有效降低实验的随机性, 对于二维、三维环境的大小不同, 分别设置 20, 10 的步长以保证解的多样性, 并合理控制迭代速度。

4.1. 二维环境仿真

首先在二维环境中进行两种算法的对比, 地图环境设置为边长为 400 的二维平面, 将机械臂简化为一个质点, 设置起始点位于 (1, 1), 目标点位于 (399, 399)。两种算法的模拟结果如下图所示。

从图 3 中观察发现, RRT 算法扩展过程缓慢, 改进后的 RRT 算法在扩展过程中的效率明显变高, 运算量有所减少, 改进 RRT 算法在二维空间的情况下有着一个比较优越的性能。

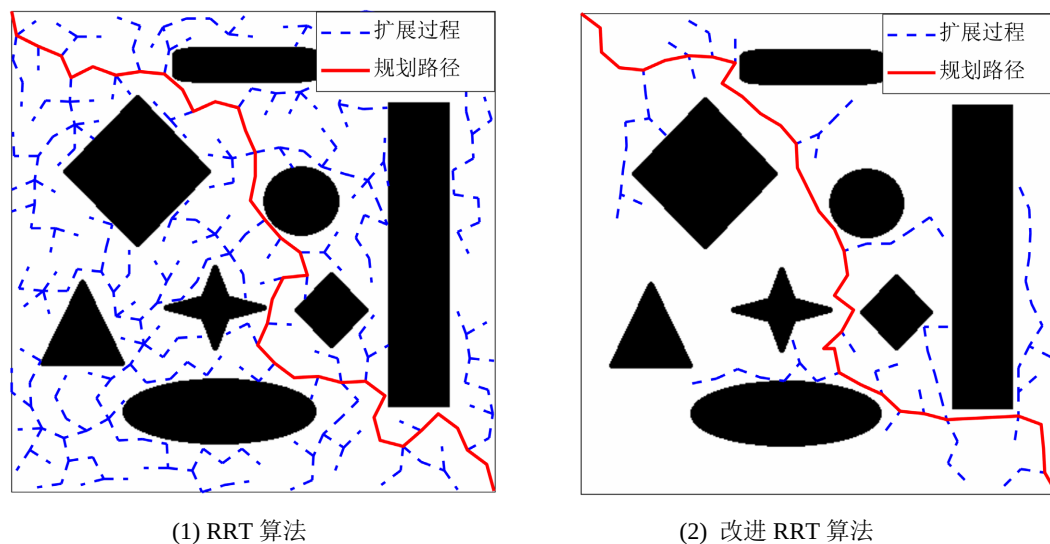


Figure 3. Comparison of RRT with its improved algorithm in 2D environment

图 3. 二维环境下的 RRT 与其改进算法的比较

4.2. 三维环境仿真

接下来在三维的环境中测试改进 RRT 算法的性能，图中的绿点表示为起始点，设置为(10, 10, 0)，蓝色的点为目标点，设置为(140, 140, 140)。三维环境下的 RRT 算法与改进 RRT 算法的性能比较如图 4 所示。

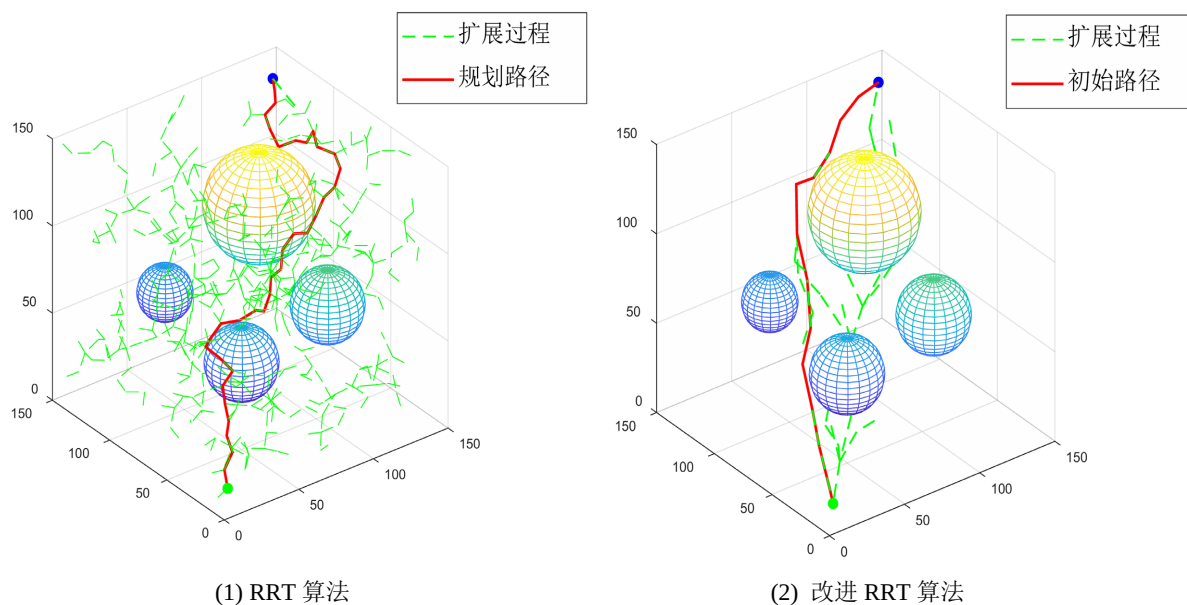


Figure 4. Comparison of RRT algorithm with improved RRT algorithm in 3D environment

图 4. 三维环境下的 RRT 算法与改进 RRT 算法的比较

在三维仿真环境中，RRT 算法规划的路径由于算法的自身局限性产生了较多的分支，冗余采样点多，规划过程耗时较长。而改进 RRT 算法能够在短时间找到代价较小的路径，也发现改进 RRT 算法在三维

Table 2. Performance comparison between RRT algorithm and improved RRT algorithm in 3D environment
表 2. 三维环境下 RRT 算法与改进 RRT 算法的性能对比

算法	路径长度/mm	采样点数	路径时间/s
RRT	916.34	664	4.89
改进 RRT	719.39	232	1.82

环境下要比二维环境下的效果更为优越，这也保证了机械臂在实际工作环境中拥有比较出色的性能。三维环境下的 RRT 算法与改进 RRT 算法的性能比较结果如表 2 所示。

5. 结论

目前对于机械臂路径规划仍旧严重依赖人工示教，不利于工业生产中工作效率的提升。为了解决这个问题，本文基于对路径规划算法的研究，在以 RRT 算法的基础上，针对其规划路径长，规划速度慢的缺点进行改进，提出了一种改进 RRT 算法来实现路径规划。基于 MATLAB 的一系列仿真证明改进 RRT 算法在规划路径时间和路径长度上都有明显的优越性，最后用模拟仿真实验证明改进 RRT 算法的可行性，将算法的规划路径缩短了 21.5%，规划时间减少了 62.8%，采样点数量减少了 65.1%，大幅提升了算法的性能，也证明改进 RRT 算法在工业机械臂控制中具有实用价值。

参考文献

[1] Shubham, S., Shukla, N.K. and Vibhav, K.S., *et al.* (2019) Multi Robot Path Planning Based on Cuckoo Search and with PSO Algorithm in the Complex and the Unknown Environment. *Carbohydrate Research*, **8**, 2924-2923.

[2] Hart, P., Nilsson, N. and Raphael, B. (1968) A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions on Systems Science and Cybernetics*, **4**, 100-107. <https://doi.org/10.1109/tssc.1968.300136>

[3] Jin, R., Rocco, P. and Geng, Y. (2021) Cartesian Trajectory Planning of Space Robots Using a Multi-Objective Optimization. *Aerospace Science and Technology*, **108**, 106360. <https://doi.org/10.1016/j.ast.2020.106360>

[4] Kavraki, L.E., Kolountzakis, M.N. and Latombe, J.-C. (1998) Analysis of Probabilistic Roadmaps for Path Planning. *IEEE Transactions on Robotics and Automation*, **14**, 166-171. <https://doi.org/10.1109/70.660866>

[5] Kuffner, J. and LaValle, S.M. (2000) RRT-Connect: An Efficient Approach to Single-Query Path Planning. In *Proceedings of the IEEE International Conference on Robotics and Automation*, San Francisco, 24-28 April 2000, 1-7.