

机器人路径规划中快速扩展随机树算法的改进研究

王硕^{1,2}, 段蓉凯^{1,2}, 廖与禾^{1,2}

(1. 西安交通大学现代设计及转子轴承系统教育部重点实验室, 710049, 西安;

2. 西安交通大学陕西省机械产品质量保障与诊断重点实验室, 710049, 西安)

摘要: 针对基本快速扩展随机树(RRT)算法在路径规划中具有树的扩展随机性大、冗余节点多、容易在目标点周围发生振荡、规划的路径较长等问题,提出了一种改进的 RRT 算法。该算法首先采用目标偏向策略,通过引入动态权重系数使树尽可能地在向目标点进行扩展的同时又能够即时地避开障碍物;利用自适应扩展步长减少树在目标点附近的振荡;最后,对路径进行剪枝处理,并用三次 B 样条曲线对剪枝后的路径进行平滑处理。仿真分析的结果表明,与基本 RRT 算法相比,改进的 RRT 算法有效减少了冗余节点数,规划的路径更短,减少了 19.56%,同时规划时间大大降低,减少了 54.08%,有效地提高了路径规划的效率。

关键词: RRT 算法;动态权重系数;自适应扩展步长;剪枝;平滑处理

中图分类号: TP306.1 **文献标志码:** A

DOI: 10.7652/xjtuxb202207001 **文章编号:** 0253-987X(2022)07-0001-08



OSID 码

Research on Improvement of Rapidly Exploring Random Tree Algorithm in Robot Path Planning

WANG Shuo^{1,2}, DUAN Rongkai^{1,2}, LIAO Yuhe^{1,2}

(1. Key Laboratory of Education Ministry for Modern Design and Rotor-Bearing System, Xi'an Jiaotong University,

Xi'an 710049, China; 2. Shaanxi Key Laboratory of Mechanical Product Quality Assurance and Diagnostics,

Xi'an Jiaotong University, Xi'an 710049, China)

Abstract: In order to solve the problems of the basic rapidly exploring random tree (RRT) algorithm in path planning, such as large randomness, excessive redundant nodes, oscillation around the target point and long planned paths, an improved RRT algorithm was proposed. Firstly, the algorithm applies the target bias strategy and introduces dynamic weight coefficient to make the tree expand to the target point as much as possible while avoiding obstacles instantly. Next, the oscillation of the tree near the target point is reduced by using the adaptive expansion step. Finally, the path is pruned and smoothed by cubic B-spline curves. Simulation results show that compared with the basic RRT algorithm, the improved RRT algorithm effectively reduces the number of redundant nodes, the planned path is shorter by 19.56%, and the planning time is greatly reduced by 54.08%, which effectively improves the efficiency of path planning.

Keywords: RRT algorithm; dynamic weight coefficient; adaptive expansion step; pruning; smoothing processing

随着社会和科学的不断进步,机器人技术得到了迅速的发展,智能机器人逐渐在医疗卫生、家庭服务、军事航天、工业生产等领域得到广泛的应用。与此同时,路径规划作为机器人技术中重要的研究领域,也开始迅速发展^[1-4]。

路径规划的研究目的是在有障碍物的环境中,使机器人可通过算法自主规划出无碰撞的、从起点到目标点的可行路径,或满足某些约束条件的最优路径^[5]。常用的算法有图搜索法^[6]、栅格法^[7]、人工势场法^[8]、遗传算法^[9]、蚁群算法^[10]、RRT法^[11]等。其中,RRT算法因具有泛用性好、适合移动机器人和多自由度工业机器人等特点而被广泛使用,但由于采样过程的随机性,基本RRT算法存在树的扩展随机性大、冗余节点多、容易在目标点周围发生振荡、规划的路径较长等问题。

为解决基本RRT算法所存在的问题,国内外学者进行了大量有针对性的研究工作。例如,文献[12]在基本RRT算法的基础上提出了双向随机搜索(从起点和终点同时扩展两颗树)的RRT-Connect算法,减少了路径规划的时间;文献[13]提出了引入贪婪策略的RRT-Connect算法,使树尽可能地目标点方向生长,搜索路径更优;文献[14]提出了动态调整父节点从而对路线进行优化的RRT-star算法,缩短了规划路径的长度;文献[15]提出了Quick-RRT-star算法,扩大了RRT-star算法动态调整父节点的范围,对搜索路径进行了优化;文献[16]以一定的概率将目标点作为采样点,文献[17]引入了权重系数,增加了树向目标点方向的扩展分量,文献[18]引入了引力系数,使树在扩展时一直受到目标点方向的引力影响,这几种方法都能使树偏向目标点方向进行生长,从而减少了树扩展时的随机性。

这些研究在机器人路径规划方面进行了许多有益的探索,并取得了一定的进展。但目前对于树扩展随机性大的问题的改进研究大多只考虑了如何让树偏向目标点方向进行生长,而没有考虑障碍物的影响,因此造成树始终偏向目标点方向生长,无法根据障碍物信息灵活调整扩展方向并在障碍物附近产生大量无效节点的问题。此外,目前对于RRT算法容易在目标点附近发生振荡问题的研究也还存在不足。为进一步解决这些问题,本文提出了一种改进的RRT算法。首先,通过引入动态权重系数以提高树扩展时的灵活性;其次,针对扩展树容易在目标点附近产生振荡的问题,研究了扩展步长的自适

应设置方法。最后,结合剪枝处理^[19]和基于三次B样条曲线^[20]的平滑处理完成了规划路径的优化。

1 RRT算法基本原理

基本RRT算法的扩展图如图1所示。

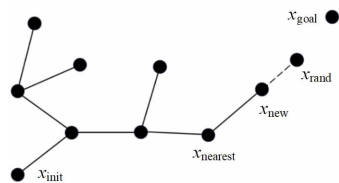


图1 基本RRT算法扩展图

Fig. 1 Basic RRT algorithm extension diagram

该算法以起点为根节点,在状态空间中进行随机采样,通过特定的约束条件进行随机树的扩展。当随机树上的点到达目标点设定的邻域范围内时,即可找到一条从起点到目标点的唯一路径。

RRT算法的基本步骤如下。

(1)设在已知的空间环境中,有定义的起点 x_{init} 和目标点 x_{goal} 。

(2)在空间中随机生成一个点 x_{rand} 并遍历已生成的扩展树,找到距离 x_{rand} 最近的节点 $x_{nearest}$,沿着 x_{rand} 向 $x_{nearest}$ 方向扩展一定的距离 e 得到新节点 x_{new} ,连接 $x_{nearest}$ 和 x_{new} 得到直线 L ;检测直线 L 是否和障碍物发生碰撞,如果未发生碰撞就将 x_{new} 加入树中。

(3)重复步骤(2)直到新节点到达目标点 x_{goal} 的邻域范围内,连接 x_{new} 和 x_{goal} ,从 x_{new} 开始依次寻找并连接父节点,即完成起点到目标点的路径规划。

基本的RRT算法存在以下缺点^[21-23]。

(1)树扩展时随机性较大:由于随机采样点是根据均匀函数产生的,导致新节点的生成缺乏偏向性。

(2)冗余节点较多:树扩展时的随机性使得RRT算法会在非目标方向的区域生成大量冗余节点,从而产生不合理的规划路径。

(3)树容易在目标点周围发生振荡:为了提高树扩展的效率,树的扩展步长总是大于目标点邻域,这导致树扩展到目标点附近时容易发生振荡现象。

(4)路径平滑性不足:节点与节点之间通过直线连接,所规划路径的各段直线之间因此存在拐角,缺乏平滑性,难以满足机器人运动的平稳性要求。

2 改进RRT算法

基于上述对基本RRT算法和相关改进研究存

在问题的分析,本文首先提出采用动态权重系数和自适应扩展步长的方式优化路径搜索过程,并在对初步规划后的路径进行剪枝处理以去除冗余节点后,利用平滑处理完成路径的最终规划,具体详述如下。

2.1 动态权重系数

基本 RRT 算法在进行树的扩展时没有考虑目标点的引导信息,因此生成了大量远离目标区域的无效节点,导致路径规划时具有较大的随机性。文献[17]提出了在目标点和随机点两个方向分别取固定权重系数 p_1 和 p_2 (p_1 和 p_2 的最优值由仿真实验确定),再进行矢量合成确定扩展方向和扩展距离的方法。这种方法增加了树扩展时向目标点的偏向性,能够明显地降低路径规划的时间,但是 p_1 总是取大于 p_2 的固定值导致遇到障碍物时容易产生大量无效节点,这就增加了路径规划所需的迭代次数。针对这一问题,本文引入了动态权重系数,使树尽可能地在向目标点进行扩展的同时又能够即时地避开障碍物。为了提高算法对障碍物的敏感性,本文将目标点方向和随机点方向的权重系数取为 p 和 $1-p$ 这两个相互制约的参数。不考虑权重系数的变化,引入权重系数后树的扩展示意图如图 2 所示。

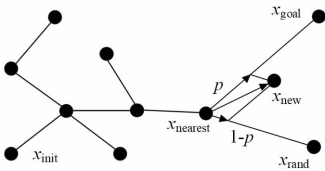


图 2 引入权重系数后树的扩展示意图

Fig. 2 Expansion diagram after introducing weight coefficients

新节点 x_{new} 的坐标由以下公式决定

$$x_{new} = x_{nearest} + p \frac{x_{goal} - x_{nearest}}{\|x_{goal} - x_{nearest}\|} e + (1-p) \frac{x_{rand} - x_{nearest}}{\|x_{rand} - x_{nearest}\|} e \quad (1)$$

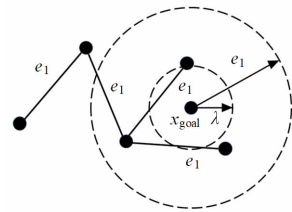
式中: p 代表权重系数, $0 < p < 1$ 。当 $p > 0.5$ 时,新节点将偏向目标点方向;当 $p < 0.5$ 时,新节点将偏向随机方向。

由于路径搜索开始前障碍物的信息未知,路径规划过程中 p 的取值如可根据障碍物的情况自适应地动态调整,对于路径规划而言是更合理的方式。文献[24]提出一种将人工势场法与 RRT 算法相结合的方法。该方法与动态权重系数法效果相似,只是由于需要在生成新节点时不断地更新引力和斥力这两个参数,计算效率并不理想。本文在上述工作

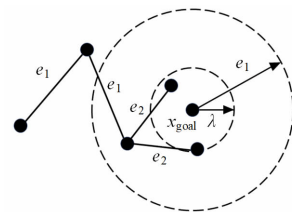
的基础上,进一步提出一种 p 值动态调整的思路,即首先考虑目标点方向因素占主导的情况 ($p > 0.5$),使树偏向目标点进行扩展。如果得到的 x_{new} 与 $x_{nearest}$ 之间的连线 L 与障碍物没有发生碰撞,则将 x_{new} 加入扩展树。否则,考虑随机方向因素占主导的情况并调整 p 值使 $p < 0.5$,以便树能快速避开障碍物。如果新的 x_{new} 与 $x_{nearest}$ 之间的连线 L 与障碍物没有发生碰撞,则将 x_{new} 加入扩展树。依此进入下一轮迭代。这种动态调整 p 值的方式让树能够在偏向目标点扩展的同时,快速地避开障碍物。

2.2 自适应扩展步长

新节点 x_{new} 生成后会判断该节点和目标点 x_{goal} 之间的距离是否小于目标点的邻域 λ ,若满足条件就将 x_{new} 和 x_{goal} 相连。一般情况下,为了确保树扩展时的效率,扩展步长 e 总是大于 λ ,当 $x_{nearest}$ 和 x_{goal} 之间的距离为 (λ, e) 范围内的值时,如果树继续以步长 e 进行扩展,容易造成新的节点 x_{new} 在目标点周围振荡,陷入局部死循环,不仅增加路径规划的时间,而且使路径在目标点附近较为曲折,增加了路径的长度和路径复杂度。为解决这一问题,本文中 e 将根据扩展树与目标点之间的距离自适应地进行调整。记扩展步长的初始设定值为 e_1 ,当 $x_{nearest}$ 和 x_{goal} 之间的距离大于 e_1 时, $e = e_1$; 当 $x_{nearest}$ 和 x_{goal} 之间距离为 $[\lambda, e_1]$ 范围内的值时, $e = e_2 = \|x_{goal} - x_{nearest}\|$ 。固定扩展步长和自适应扩展步长的扩展示意图如图 3 所示(示意图仅显示扩展步长的对比,未考虑动态权重系数)。



(a) 固定扩展步长扩展示意图



(b) 自适应扩展步长扩展示意图

图 3 固定和自适应扩展步长的扩展示意图

Fig. 3 Expansion diagram of fixed and adaptive expansion steps

2.3 路径剪枝和平滑处理

引入动态权重系数后树扩展的随机性大大减小,但是由于生成的节点众多,依然存在冗余节点。本文对初步路径规划得到的节点集合进行剪枝,从起点 x_{init} 开始尽量去除集合中不必要的节点。

剪枝处理的具体思路如下:对初步路径规划得到的节点集合,从起点 x_{init} 开始,依次遍历后续节点 x_k ,如果 x_{init} 和 x_k 之间的连线和障碍物之间没有发生碰撞,则将 x_{init} 和 x_k 之间的所有节点都删除;如果 x_{init} 和 x_k 之间的连线和障碍物之间发生了碰撞,则从节点 x_{k-1} 开始重复以上步骤,直到遍历到目标点 x_{goal} ,最后将剩余的节点结合依次进行连接,得到新的路径。这种剪枝方式极大地减少了路径上的节点数量。

剪枝后路径的拐点位置仍然是分段的,不平滑。机器人按照这样的路径行走,会在拐点处因为行进方向变化产生冲击,影响机器人的运行平稳性。因此需要进一步对路径进行平滑处理。鉴于 B 样条曲线精度较高,能够较好地控制曲线的平滑度^[25],且三次 B 样条拟合曲线的曲率半径较小,在拐点的过渡较为平滑,计算量也相对较少,本文选用三次 B 样条曲线对剪枝后的路径进行平滑处理。

K 次 B 样条曲线可表示为

$$p(u) = \sum_{i=0}^n d_i B_{i,K}(u), \quad (i = 0, 1, 2, \dots, n) \quad (2)$$

其中, d_i 为曲线控制的顶点,即树的节点; $B_{i,K}$ 为 K 次 B 样条曲线的基函数。K 次分段曲线由节点向量 $U = [u_0, u_1, \dots, u_{n+K+1}]$ 决定。三次 B 样条曲线的基函数为

$$\begin{cases} B_{0,3} = \frac{1}{6}(-u^3 + 3u^2 - 3u + 1) \\ B_{1,3} = \frac{1}{6}(3u^3 - 6u^2 + 4) \\ B_{2,3} = \frac{1}{6}(-3u^3 + 6u^2 + 3u + 1) \\ B_{3,3} = \frac{1}{6}u^3 \end{cases} \quad (3)$$

2.4 改进 RRT 算法流程

综合以上几种改进措施,改进的 RRT 算法流程图如图 4 所示。

改进的 RRT 算法流程如下。

(1)在目标空间中,设定起点 x_{init} 和目标点 x_{goal} ,扩展步长 e 的初始值 e_1 ,目标点的邻域 λ 。

(2)生成随机采样点 x_{rand} ,遍历扩展树找到最近节点 $x_{nearest}$,检测 $x_{nearest}$ 和 x_{goal} 之间的距离是否大

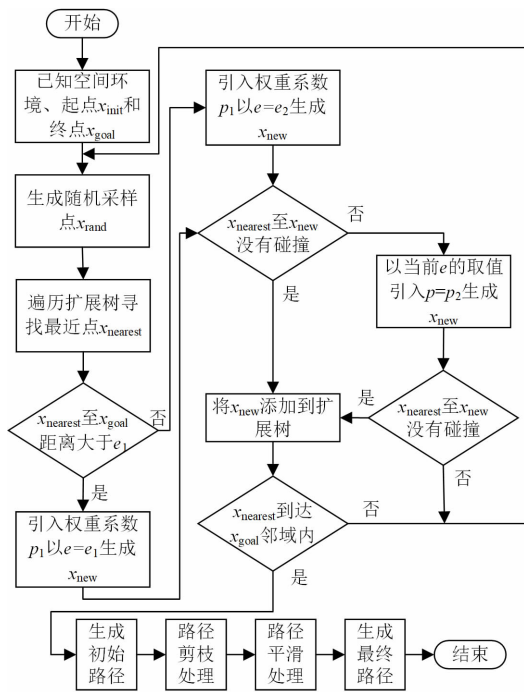


图4 改进的 RRT 算法流程图

Fig. 4 The flow chart of improved RRT algorithm

于 e_1 。

(3)如果步骤(2)条件满足,则在 e_1 的基础上引入权重系数 p_1 得到新节点 x_{new} ,否则将 e 的值设为新值 e_2 ,在 e_2 的基础上引入权重系数 p_1 得到新节点 x_{new} 。

(4)检测 x_{new} 和 $x_{nearest}$ 之间的连线 L 是否和障碍物发生碰撞,如果未发生碰撞,就将 x_{new} 加入扩展树,否则在当前 e 取值的基础上引入权重系数 p_2 得到新节点 x_{new} 。

(5)检测 x_{new} 点和 $x_{nearest}$ 之间的连线 L 是否和障碍物发生碰撞,如果未发生碰撞,就将 x_{new} 加入扩展树,否则返回步骤(2)。

(6)检测 x_{init} 和 x_{goal} 之间的距离是否小于 λ ,如果满足条件,连接 x_{new} 和目标点,从 x_{new} 开始依次寻找并连接父节点得到起点到目标点的路径。

(7)对生成的初始路径进行剪枝处理和曲线拟合。

3 仿真实验

3.1 初始路径规划仿真

为了对改进的 RRT 算法的效果进行验证,在 matlab 中进行了二维空间环境下的仿真。设计的地图包含较多障碍物,以验证改进的算法能够快速避开障碍物。环境场景的大小为 $800 \text{ mm} \times 800$

mm,设定起点坐标(50 mm,50 mm),目标点坐标(750 mm,750 mm),初始扩展步长 $e_1=50$ mm,目标点邻域 $\lambda=20$ mm,分别使用 RRT 算法、RRT-star 算法、固定权重系数的 RRT 算法(引入自适应扩展步长)和本文改进的 RRT 算法进行路径规划对比分析。为了确保树的偏向性,固定权重系数的 RRT 算法的权重系数 p 取值为 0.8,本文改进的 RRT 算法的权重系数 p 取值为 0.8(目标点方向因素占主导时)和 0.2(随机方向因素占主导时)。黑色粗线代表规划出的路径,四种算法的规划结果如图 5 所示。

表 1 为 4 种算法在较多障碍物环境下分别进行 500 次路径规划所得的平均数据。数据表明,在路径规划时间和路径长度这两个主要指标上,改进的 RRT 算法相较基本 RRT 算法和 RRT-star 算法都有明显的优势。改进的 RRT 算法和基本 RRT 算法相比,路径规划时间减少 54.08%,路径长度减少

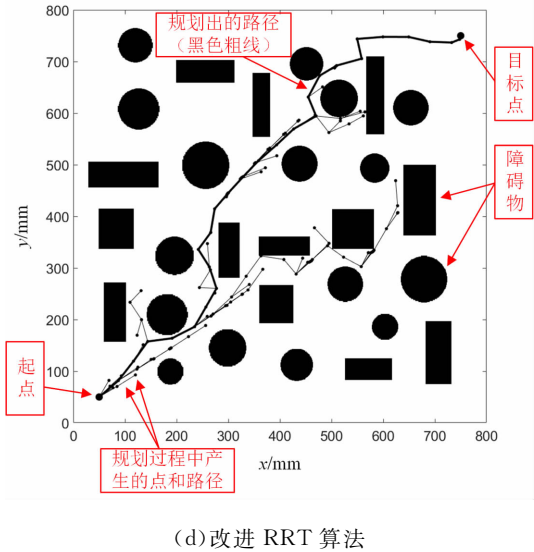
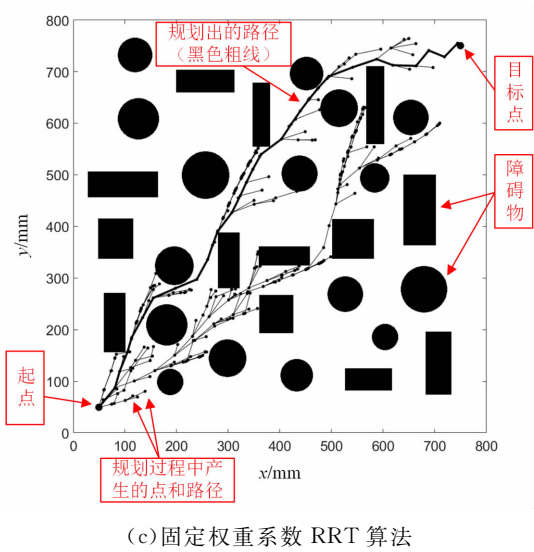
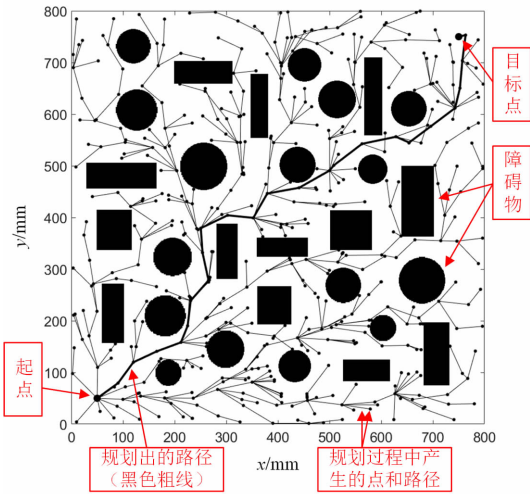
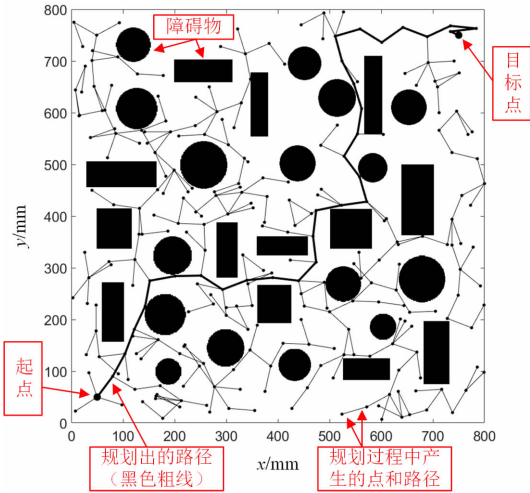


图 5 4 种 RRT 算法的路径规划结果
Fig. 5 Path planning results of four RRT algorithms

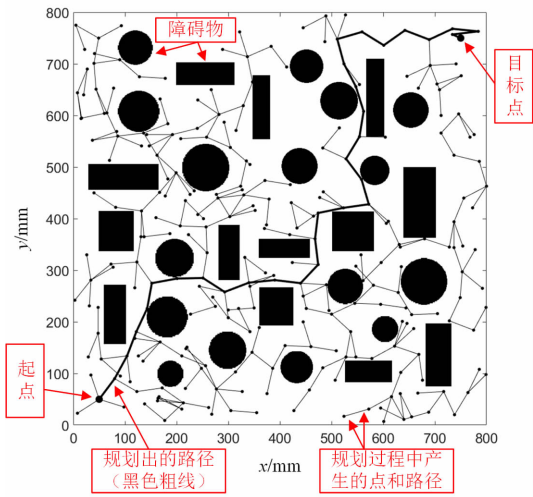


19.56%;改进的 RRT 算法和 RRT-star 算法相比,路径长度的减小虽然不甚明显,但所需的路径规划时间大幅减少。改进的 RRT 算法相较使用固定权重系数的 RRT 算法,虽然路径长度略微增加了 5.21%,但是路径规划时间大幅减少了 60.31%,算法的改进是有效的。

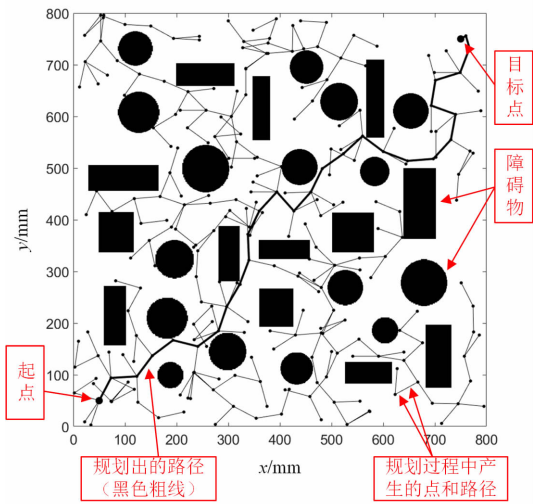
Table 1 Path planning data of four algorithms		
算法	路径规划时间/s	路径长度/mm
基本 RRT	3.31	1 415.63
RRT-star	128.41	1 204.83
固定权重系数的 RRT	3.83	1 082.34
改进的 RRT	1.52	1 138.71

3.2 自适应扩展步长仿真

为进一步检验动态权重系数和自适应扩展步长的有效性,继续在保持仿真环境和其他参数不变的条件下,对仅引入自适应扩展步长的改进算法进行检验,仿真分析结果如图 6 所示。可以看到,基本 RRT 算法在目标点 x_{goal} 附近发生了振荡(图 6(a)),而引入自适应扩展步长的 RRT 算法会根据 $x_{nearest}$ 和目标点之间的距离动态地调整扩展步长,有效地减少了振荡并迅速到达路径目标点(图 6(b))。



(a)基本 RRT 算法



(b)仅引入自适应扩展步长的 RRT 算法

图 6 两种 RRT 算法的路径规划结果

Fig. 6 Path planning data of two algorithms

表 2 为基本 RRT 算法、仅引入自适应扩展步长的 RRT 算法和改进的 RRT 算法在较多障碍物环境下分别进行 500 次路径规划所得到的平均数据。数据表明,引入自适应扩展步长的 RRT 算法和基本 RRT 算法相比有明显提升,改进是有效的。

引入自适应扩展步长的 RRT 算法和基本 RRT 算法相比,路径规划时间减少 34.74%,路径长度减少 2.08%。进一步的数据分析还表明,仅考虑自适应扩展步长的 RRT 算法在路径规划时间和路径长度这两个指标上与改进的 RRT 算法相比均仍有较大差距。因此,在路径规划中同时考虑动态权重系数和自适应扩展步长有其重要性。

表 2 3 种算法的路径规划数据

Table 2 Path planning data of three algorithms

算法	路径规划时间	路径长度
	/s	/mm
基本 RRT	3.31	1 415.63
RRT(仅引入自适应扩展步长)	2.16	1 386.17
改进的 RRT	1.52	1 138.71

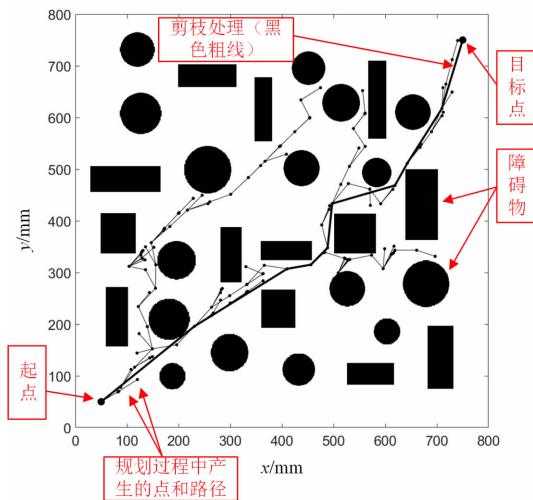
3.3 剪枝及平滑处理仿真

为了减少冗余节点并提高路径的平滑性,将初步规划后的路径进行剪枝并采用三次 B 样条进行平滑处理,matlab 中的仿真结果如图 7 所示。

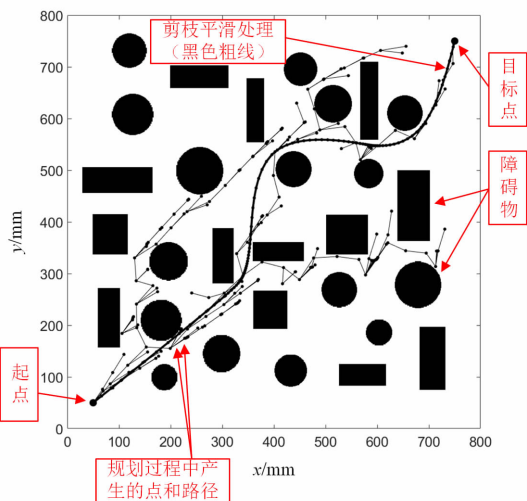
从图 7 中可以看出,剪枝后的路径节点和拐角都大大减少,平滑处理后的路径和原路径相比具有较好的平滑性,能够更好地满足机器人运动的平稳性要求。

4 结 论

针对基本 RRT 算法和相关改进研究存在的问题,本文针提出了一种改进的 RRT 算法,该算法通过引入动态权重系数和自适应扩展步长,减少了树扩展的随机性以及树在目标点周围的振荡现象;并对初步规划后的路径进行剪枝和三次 B 样条平滑



(a)剪枝处理



(b) 平滑处理

图 7 路径平滑和剪枝处理

Fig. 7 Path smoothing and pruning

处理。本算法在多处对基本 RRT 算法进行了改进,具有一定的参考价值。

根据仿真结果,与基本 RRT 算法相比,本算法的有效性和优越性主要体现在以下几个方面:

(1)与基本 RRT 算法相比,改进的 RRT 算法进行路径规划所需的时间大幅降低,在仿真环境下,路径规划时间减少 54.8%;

(2)与基本 RRT 算法相比,改进的 RRT 算法规划的路径长度减少,在仿真环境下,路径长度减少了 19.56%;

(3)将改进的 RRT 算法初步规划得到的路径进行剪枝和平滑处理后,新的路径具有较好的平滑性,能够更好地满足机器人运动的要求。

参考文献:

[1] 陈乐瑞,曹建福,胡河宇.采用非线性频谱的重载工业机器人故障机理分析[J].西安交通大学学报,2021,55(5):92-101.
CHEN Lerui, CAO Jianfu, HU Heyu. Fault mechanism analysis for heavy-load industrial robot with non-linear spectrum [J]. Journal of Xi'an Jiaotong University, 2021, 55(5): 92-101.

[2] 蒋一辰,陈刚.融合自适应曲线预瞄的驾驶机器人转向操纵粒子群优化滑模控制[J].西安交通大学学报,2021,55(3):175-185.
JIANG Yichen, CHEN Gang. Particle swarm optimization sliding mode steering control of driving robot integrated with adaptive curve preview [J]. Journal of Xi'an Jiaotong University, 2021, 55(3): 175-185.

[3] 鞠兴全,毛新涛,宰守刚,等.协作机器人的发展及在航天领域应用分析[J].中国航天,2022(2):63-67.
JU Xingquan, MAO Xintao, ZAI Shougang, et al. Development and application analysis of cooperative robot in aerospace field [J]. Aerospace China, 2022(2): 63-67.

[4] 穆小奇,张小栋,徐海鹏,等.防老年人跌倒的助老机器人人机系统动态稳定性分析模型[J].西安交通大学学报,2020,54(8):67-76.
MU Xiaoqi, ZHANG Xiaodong, XU Haipeng, et al. Dynamic stability analysis models for human-robot systems of elderly-assistant robots helping elderly prevent falling [J]. Journal of Xi'an Jiaotong University, 2020, 54(8): 67-76.

[5] AJEIL F H, IBRAHEEM I K, AZAR A T, et al. Grid-based mobile robot path planning using aging-based ant colony optimization algorithm in static and dynamic environments [J]. Sensors, 2020, 20(7): 1880.

[6] ZHONG Xunyu, TIAN Jun, HU Huosheng, 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.

[7] 魏彤,龙琛.基于改进遗传算法的移动机器人路径规划[J].北京航空航天大学学报,2020,46(4):703-711.
WEI Tong, LONG Chen. Path planning for mobile robot based on improved genetic algorithm [J]. Journal of Beijing University of Aeronautics and Astronautics, 2020, 46(4): 703-711.

[8] 赵晓,王铮,黄程侃,等.基于改进 A* 算法的移动机器人路径规划[J].机器人,2018,40(6):903-910.
ZHAO Xiao, WANG Zheng, HUANG Chengkan, et al. Mobile robot path planning based on an improved A* algorithm [J]. Robot, 2018, 40(6): 903-910.

[9] LIANG Ye, WANG Lindong. Applying genetic algorithm and ant colony optimization algorithm into marine investigation path planning model [J]. Soft Computing, 2020, 24(11): 8199-8210.

[10] 刘贵杰,刘鹏,穆为磊,等.采用能耗最优改进蚁群算法的自治水下机器人路径优化[J].西安交通大学学报,2016,50(10):93-98.
LIU Guijie, LIU Peng, MU Weilei, et al. A path optimization algorithm for AUV using an improved ant colony algorithm with optimal energy consumption [J]. Journal of Xi'an Jiaotong University, 2016, 50

- (10): 93-98.
- [11] WANG Jiankun, CHI Wenzheng, LI Chenming, et al. Neural RRT*: learning-based optimal path planning [J]. IEEE Transactions on Automation Science and Engineering, 2020, 17(4): 1748-1758.
- [12] KUFFNER J J, LAVALLE S M. RRT-connect: an efficient approach to single-query path planning [C]// Proceedings of 2000 IEEE International Conference on Robotics and Automation vol. 2. Piscataway, NJ, USA: IEEE, 2000: 995-1001.
- [13] 李磊, 吴翔, 包祥威, 等. 基于改进 RRT-Connect 算法的机械臂抓取路径规划 [J]. 机器人技术与应用, 2021(2): 28-32.
- LI Lei, WU Xiang, BAO Xiangwei, et al. Grasping path planning of manipulator based on improved RRT-Connect algorithm [J]. Robot Technique and Application, 2021(2): 28-32.
- [14] KARAMAN S, FRAZZOLI E. Incremental sampling-based algorithms for optimal motion planning [J]. The International Journal of Robotics Research, 2011, 30(7): 846-894.
- [15] JEONG I B, LEE S J, KIM J H. Quick-RRT*: triangular inequality-based implementation of RRT* with improved initial solution and convergence rate [J]. Expert Systems with Applications, 2019, 123: 82-90.
- [16] 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/OL]. 计算机集成制造系统 [2021-09-03]. <http://kns.cnki.net/kcms/detail/11.5946.TP.20211022.1559.022.html>.
- LI Yang, ZHANG Lei, LI Pengfei, et al. Motion planning method of manipulator based on improved RRT combined with B-spline [J/OL]. Computer Integrated Manufacturing Systems [2021-09-03]. <http://kns.cnki.net/kcms/detail/11.5946.TP.20211022.1559.022.html>.
- [17] 李季, 史晨发, 邵磊, 等. 基于改进 RRT 算法的 6-DOF 机器人路径规划 [J]. 计算机应用与软件, 2020, 37(9): 221-226.
- LI Ji, SHI Chenfa, SHAO Lei, et al. Path planning of 6-DOF robot based on improved RRT algorithm [J]. Computer Applications and Software, 2020, 37(9): 221-226.
- [18] 刘恩海, 高文斌, 孔瑞平, 等. 改进的 RRT 路径规划算法 [J]. 计算机工程与设计, 2019, 40(8): 2253-2258.
- LIU Enhai, GAO Wenbin, KONG Ruiping, et al. Improved RRT path planning algorithm [J]. Computer Engineering and Design, 2019, 40(8): 2253-2258.
- [19] 张永贵, 李海龙. 铣削机器人去除锻件裂纹的轨迹规划研究 [J]. 组合机床与自动化加工技术, 2018(7): 1-4, 9.
- ZHANG Yonggui, LI Hailong. Trajectory planning research for removing forging crack with milling robot [J]. Modular Machine Tool & Automatic Manufacturing Technique, 2018(7): 1-4, 9.
- [20] 刘成菊, 韩俊强, 安康. 基于改进 RRT 算法的 RoboCup 机器人动态路径规划 [J]. 机器人, 2017, 39(1): 8-15.
- LIU Chengju, HAN Junqiang, AN Kang. Dynamic path planning based on an improved RRT algorithm for RoboCup robot [J]. Robot, 2017, 39(1): 8-15.
- [21] 刘晓倩, 张辉, 王英健. 基于改进 RRT 的路径规划算法 [J]. 自动化技术与应用, 2019, 38(5): 96-100.
- LIU Xiaoqian, ZHANG Hui, WANG Yingjian. Improved path planning algorithm of RRT [J]. Techniques of Automation and Applications, 2019, 38(5): 96-100.
- [22] 荆泽成, 张秋菊, 杨瑞. 基于改进 RRT 的核退役机器人避障方法 [J]. 现代制造工程, 2022(2): 46-51.
- JING Zecheng, ZHANG Qiuju, YANG Rui. Obstacle avoidance method for nuclear decommissioning robot based on improved RRT [J]. Modern Manufacturing Engineering, 2022(2): 46-51.
- [23] 张建冬, 王东, 马立东, 等. 基于改进 RRT 算法的移动机械臂路径规划 [J]. 电子测量技术, 2021, 44(23): 48-53.
- ZHANG Jiandong, WANG Dong, MA Lidong, et al. Path planning of mobile manipulator based on improved RRT algorithm [J]. Electronic Measurement Technology, 2021, 44(23): 48-53.
- [24] 谢龙. 冗余机械臂动态避障规划 [D]. 杭州: 浙江大学, 2018.
- [25] 李金良, 舒翰儒, 刘德建, 等. 基于改进 RRT 路径规划算法 [J]. 组合机床与自动化加工技术, 2021(2): 22-24, 29.
- LI Jinliang, SHU Hanru, LIU Dejian, et al. Path planning algorithm based on improved RRT [J]. Modular Machine Tool & Automatic Manufacturing Technique, 2021(2): 22-24, 29.

(编辑 荆树蓉 杜秀杰)