

# 动态步长的 RRT 路径规划算法

王道威,朱明富,刘 慧

(华中科技大学 自动化学院,湖北 武汉 430074)

**摘 要:**传统的快速扩展随机树(RRT)算法虽然有很多优良特性,但是由于扩展点的随机选取,规划出来的路径具有很大的随机性。文中在对 RRT 算法改进的基础上,提出了一种动态步长的 RRT 路径规划算法。其中步长为 RRT 生长的最小单位长度。动态步长的 RRT 算法是在对传统 RRT 算法的基础上,添加了动态步长的特性,改善了快速扩展随机树的不确定性,提高了避障能力,使得算法确定性和高避障能力兼备。仿真实验结果表明,该算法在路径规划中具有路径确定、速度快和高避障能力的特点。

**关键词:**快速扩展随机树;动态步长;路径确定性;高避障能力

中图分类号:TP301.6

文献标识码:A

文章编号:1673-629X(2016)03-0105-03

doi:10.3969/j.issn.1673-629X.2016.03.025

## Rapidly-exploring Random Tree Algorithm Based on Dynamic Step

WANG Dao-wei,ZHU Ming-fu,LIU Hui

(School of Automation,Huazhong University of Science and Technology,Wuhan 430074,China)

**Abstract:**Although the traditional Rapidly-exploring Random Tree (RRT) algorithm has many good features,there is a lot of randomness in path planning of RRT because of the random selection of the vertex. Based on the improvement of RRT algorithm,a new RRT path planning algorithm of dynamic step size is proposed in this paper. The step size is the minimum unit length when RRT exploring. Based on traditional RRT,the dynamic step size is added,avoiding the uncertainty,and the obstacle avoidance capability is improved,thus the path planning of RRT algorithm has both obstacle avoidance ability and high certainty. The results of simulation experiments show that the algorithm has the features of avoiding the uncertainty,fast speed and obstacle avoidance in path planning.

**Key words:**RRT;dynamic step size;path planning certainty;obstacle avoidance

## 0 引 言

随着无人机、无人汽车等无人智能设备的不断应用和发展,RRT 路径规划算法<sup>[1-3]</sup>越来越多地被应用。RRT 算法是一种随机搜索算法,它能在保证系统实时性的前提下,寻找到问题的较优解。但是 RRT 算法具有随机性<sup>[4]</sup>,规划出来的路径偏离目标点,与理想路径相比有较大差异。

文中在经典 RRT 算法的基础上引入目标引力函数,让随机树朝着目标点方向生长,使得规划出来的路径更加接近理想路径。但是这种添加了目标引力函数的 RRT 算法虽然降低了路径的随机性,但同时也降低了算法的避障能力,起始点和目标点障碍物较多的情况下不能规划出路径。文中在 RRT 算法的基础上引入了动态步长的概念。步长即为 RRT 生长的最小单位长度。在 RRT 算法扩展新节点时,当遇到障碍物

时自动减小目标点方向的步长,没遇到障碍物时再恢复原步长。改进后的 RRT 路径规划算法兼具路径确定性和高避障能力的优点。

## 1 RRT 算法

快速随机扩展树(RRT)算法由 Steven M. LaValle 于 1998 年在文献[5]中首先提出。RRT 算法是从空间中一个起始节点出发,通过随机采样,不断增加新节点,生成一个随机扩展树。当增加的新节点中有目标节点时,起始节点到目标节点就会至少有一条通路,称为路径。RRT 算法已应用于飞行器运动规划<sup>[6]</sup>和移动机器人路径规划中<sup>[7]</sup>。

### 1.1 经典 RRT 算法

RRT 算法的作用是在度量空间  $X$  中,选择一条从

初始状态点  $x_{init}$  到目标状态点  $x_{goal}$  或者到目标状态点区域  $x_{goal} \in X$  的路径。通常选择距离空间  $C$  作为度量空间,即  $X = C$ 。在度量空间中路径无法通过的区域称为障碍区域,记为  $x_{obs} \in X$ 。障碍区域在度量空间中的补集称为自由区域,记为  $x_{free} \in X$ 。令  $x_{rand}$  为每次扩展 RRT 时在自由区域中随机选取的点,  $x_{rand} \in x_{free}$ 。令  $x_{near}$  为每次扩展 RRT 时在当前 RRT 中与  $x_{rand}$  距离最近的点。 $x_{new}$  为每次新增加的扩展点。给定一个初始状态  $x_{init}$ , 一个随机扩展树  $T$ , 生成  $K$  个顶点的方法如下:

```

GENERATE_RRT (  $x_{init}, K, \Delta t$  )
 $T$ .init(  $x_{init}$  );
For  $k = 1$  to  $K$  do
 $x_{rand} \leftarrow \text{RANDOM\_STATE}()$ ;
 $x_{near} \leftarrow \text{NEAREST\_STATE}(x_{rand}, T)$ ;
 $u \leftarrow \text{SELECT\_INPUT}(x_{rand}, x_{near})$ ;
 $x_{new} \leftarrow \text{NEW\_STATE}(x_{near}, u, \Delta t)$ ;
 $T$ .add_vertex(  $x_{new}$  );
 $T$ .add_edge(  $x_{near}, x_{new}, u$  );
Return  $T$ 

```

其中,  $\text{RANDOM\_STATE}()$  在自由空间中随机选取一点  $x_{rand}$ ,  $\text{NEAREST\_STATE}()$  在当前 RRT 中选择与  $x_{rand}$  距离最近的点  $x_{near}$ ,  $\text{SELECT\_INPUT}()$  结合  $x_{rand}$  和  $x_{near}$  得到输入  $u$ ,  $\text{NEW\_STATE}()$  得到新节  $x_{new}$ 。

## 1.2 引入目标引力思想的 RRT 算法

在文献[1]中介绍了一种引入目标引力思想的 RRT 算法,能有效减少 RRT 算法的随机性,让路径朝着目标方向扩展。

在文献[2]中介绍了一种目标偏好 RRT 算法,该算法大部分扩展过程使用随机点,以一定小概率选取目标点取代随机点,将随机树向目标区域扩展。

文献[3]中对目标偏好 RRT 算法做了进一步改进,使它应用于非完整微分约束中。

经典的 RRT 算法随机性比较明显,规划出来的路径非常曲折,不适合实际使用中的路径规划。经典 RRT 算法新节点位置的计算公式如式(1)所示。

$$x_{new} = x_{near} + \rho \frac{(x_{rand} - x_{near})}{\|x_{rand} - x_{near}\|} \quad (1)$$

其中,  $\rho$  为 RRT 生长的最小单位长度,称为步长。

加入目标引力思想的 RRT 算法新节点位置的计算公式如式(2)所示。

$$x_{new} = x_{near} + \frac{\rho_1(x_{rand} - x_{near})}{\|x_{rand} - x_{near}\|} + \frac{\rho_2(x_{goal} - x_{near})}{\|x_{goal} - x_{near}\|} \quad (2)$$

其中:  $\rho_1$  为随机点方向的步长;  $\rho_2$  为目标点方向的步长。

## 2 动态步长的 RRT 算法

路径规划是按照某一性能指标搜索一条从起始状态到目标状态的最优或近似最优的无碰路径<sup>[8]</sup>。动态步长的 RRT 算法能够规划出比较理想的无碰路径。

### 2.1 步长对 RRT 算法的影响

首先考虑步长在经典 RRT 路径规划算法中的作用。经典随机扩展树生成算法扩展一个新节点的过程为:首先在度量空间中随机选取一点  $x_{rand}$ , 然后在当前扩展树中(可能为空)找到与  $x_{rand}$  距离最近的点  $x_{near}$ , 在  $x_{near}$  到  $x_{rand}$  的方向上扩展一个步长的距离得到新节点  $x_{new}$  加入扩展树中。

加入目标引力思想后的 RRT 算法能有效解决路径随机性强的问题。引入目标引力思想后的 RRT 算法与经典 RRT 算法相比添加了  $x_{goal}$  方向的分量。加入目标引力思想后,步长的选择对新节点的位置会有影响。此时新节点不再在  $x_{near}$  到  $x_{rand}$  的连线上了。当  $\rho_2 > \rho_1$  时,新节点偏向于  $x_{goal}$ , 由于  $x_{goal}$  的位置通常是固定的,扩展树朝着  $x_{goal}$  方向扩展。当  $\rho_2 < \rho_1$  时,新节点偏向于  $x_{rand}$ , 由于  $x_{rand}$  的位置是随机选取的,此时扩展树的扩展方向接近 RRT 经典算法。

引入目标引力思想后的随机扩展树朝着目标点方向发展,规划出来的路径更加接近理想路径。 $\rho_2$  相对于  $\rho_1$  取的越大,扩展树越朝着  $x_{goal}$  方向扩展,规划出来的路径越接近理想路径。

以上所讨论的都在无障碍空间中规划路径,在实际的路径规划应用中,起始点和目标点之间都或多或少存在很多障碍,很难找到一个有效的路径规划方法来应对各种复杂环境<sup>[9]</sup>。规划出的路径需要避开所有障碍到达目标点。经典 RRT 算法新节点的添加都是随机方向的,能够有效地避开障碍。引入目标引力的 RRT 算法常常为了能够使规划出的路径接近理想路径,目标点方向的步长要较大于随机点方向的步长。此时如果起始点和目标点障碍物较多,那么扩展树无法绕开障碍物到达目标点。

可以得出结论:路径理想和避障能力是相互矛盾的。为了解决这对矛盾,在目标引力 RRT 算法基础上添加了动态步长的概念。当没有遇到障碍物时取  $\rho_2 > \rho_1$ , 使扩展树尽可能朝着目标点方向扩展。当遇到障碍物时取  $\rho_2 < \rho_1$ , 使扩展树朝着随机点方向扩展,因为随机点的随机性,新节点也具有随机性,从而可以高效地避开障碍物。

### 2.2 动态步长 RRT 算法的优点

经典 RRT 算法的随机性很强,规划出来的路径与理想路径相差很大,但由于扩展节点的随机性,经典 RRT 算法具有很强的避障能力。

为了规划出更加理想的路径,在经典 RRT 算法中

引入目标引力思想,让新节点朝着目标节点的方向扩展。但是这种算法的避障能力很弱,当起始点和目标点间有很多障碍的时候不能规划出路径。

动态步长 RRT 算法具有以上两种算法的优点,不仅能朝着目标节点扩展新节点,而且具有和经典 RRT 算法一样的避障能力。动态步长 RRT 算法在路径规划中更加实用。和经典 RRT 算法一样,该算法具有高效的搜索特性,使其适合解决高维空间多自由度的复杂约束下的运动规划问题,能直接应用到非完整性约束或非完整性动力学约束规划中<sup>[10]</sup>。

动态步长 RRT 算法时间复杂度低。在文献[11-13]中介绍的滚动规划算法,每次在一个窗口中规划出最优路径,时间复杂度很高。动态步长 RRT 算法在扩展每一个新节点时,只需计算一个式(2)的值。很多类似文献[4]中提出的 RRT 改进算法,在扩展一个新节点时计算若干个随机点的估价函数,选取估价函数数值最小的作为新节点。这些做法使算法时间复杂度成倍增加。因为 RRT 扩展树一般都具有非常多的节点,所以保持算法的较低的时间复杂度非常重要。

### 3 仿真实验结果分析

实验环境:Windows Embedded 8.1, Intel® Core™ i5-3317U CPU @ 1.70 GHz, 内存 8 G。

编译工具:Visual Studio 2013。

图1~5中左下角为起始节点,右上角为目标节点。图中空心圆环为障碍物,距障碍物环心40个单位长度内的区域为障碍区域。图中黑色粗线和黑色细线共同构成随机扩展树,黑色粗线连接起始点和目标点,被选为规划出的路径。

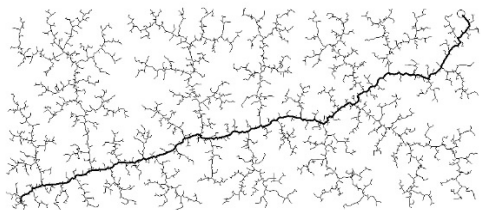


图1 经典 RRT 算法(无障碍)

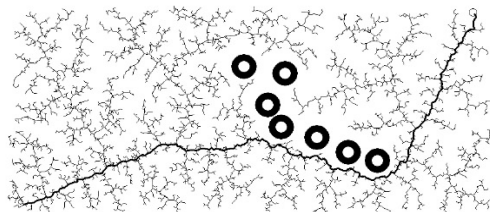


图2 经典 RRT 算法(有障碍)

在无障碍的情况下,如图1、3、4所示,都能规划出路径。由图1可见,经典 RRT 算法的随机性较强,路径比较曲折。由图3、4可见,目标引力 RRT 算法和动

态步长 RRT 算法在无障碍情况下效果相同,规划出的路径比较接近理想路径。

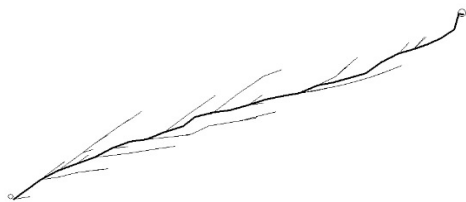


图3 目标引力 RRT 算法(无障碍)

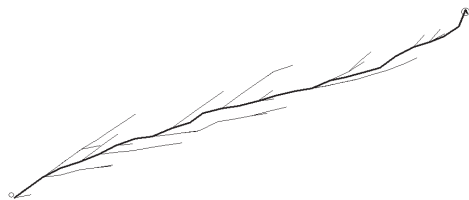


图4 动态步长 RRT(无障碍)

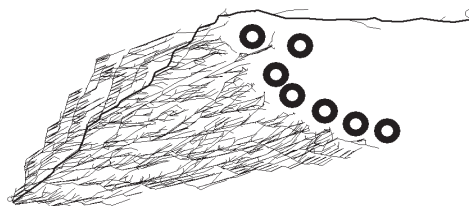


图5 动态步长 RRT(有障碍)

在有障碍的情况下,如图2、5所示,目标引力 RRT 算法无法规划出路径,其他两种算法都能规划出路径。由图2可见,经典 RRT 算法能有效避开障碍物,但是随机性较强。目标引力 RRT 算法在有障碍的情况下,由于起始点和目标点障碍物过多,随机树无法绕开障碍物到达目标节点,导致运行程序无响应。由图5可见,动态步长 RRT 算法能有效避开障碍物,而且没有经典 RRT 算法的随机性。

### 4 结束语

经典 RRT 算法具有随机搜索特性<sup>[14]</sup>,因此具有很强的避障能力,而且生成的路径不够光滑,与理想路径相差很远。目标引力思想 RRT 算法牺牲了部分随机性,规划出的路径更加光滑,但是减弱了避障能力,很多情况下无法规划出路径。动态步长的 RRT 算法不仅能够规划出接近理想路径的路径,而且具有很强的避障能力,算法时间复杂度低,能够高效广泛地应用在实际路径规划中。

#### 参考文献:

- [1] 郝利波,侯媛彬. 基于一种改进 RRT 算法的足球机器人路径规划[J]. 西安科技大学学报, 2011, 31(1): 81-85.
- [2] Urmson C, Simmons R. Approaches for heuristically biasing



从图 4 的分割结果来看,对比(e)、(g)可以看到,虽然在含噪图片进行处理时效果都不是非常理想,但文中的边缘分割效果明显好于文献[5]方法的分割效果,即文中方法也适用于含噪图片的分割。

## 4 结束语

文中以改进分水岭方法对噪声敏感以及容易造成分割的图像为目的,采用了基于小波二次去噪的分水岭算法。该方法通过对梯度图像进行二次分解重构,有效地解决了图像分割对噪声敏感的问题;然后再通过标记前景、背景的方法,使得吸水盆地只响应想要探测的目标。该方法能很好地分割无噪和含噪图片。相比一般的分水岭方法,该方法需要对梯度图像进行二次消噪并重构,所以耗费时间较多。但是改进后的方法能够较好地保留重要的目标信息,使有意义的区域能被分割出来,所以在图像分割方面具有一定的应用价值。

## 参考文献:

- [1] Zhang Y J. Image segmentation [M]. Beijing: China Science Press, 2001.
- [2] Snyder W E, Qi Hairong. Machine vision [M]. Cambridge, UK: Cambridge University Press, 2004.
- [3] Zuo Q, Shi Z K. An real-time algorithm for license plate extraction based on mathematical morphology [J]. Journal of Image and Graphics, 2003, 8(3): 281-285.
- [4] 刁智华, 赵春江, 郭新宇, 等. 分水岭算法的改进方法研究 [J]. 计算机工程, 2010, 36(17): 4-6.
- [5] 王小鹏, 罗进文. 基于形态学梯度重建的分水岭分割 [J]. 光电子·激光, 2005, 16(1): 98-101.
- [6] Gonzalez R C, Woods R E. Digital image processing [M]. 2nd

ed. Beijing: Publishing House of Electronics Industry, 2005.

- [7] 夏平, 刘馨琼, 向学军, 等. 基于形态学梯度的图像边缘检测算法 [J]. 计算机技术与发展, 2007, 17(12): 107-109.
- [8] Dai Q Y, Yu Y L. A kind of segmentation method of vehicle-license-plate images based on wavelet and mathematical morphology [J]. Journal of Image and Graphics, 2000, 5(5): 411-415.
- [9] 张德丰. MATLAB 数字图像处理 [M]. 北京: 电子工业出版社, 2010.
- [10] Kronland-Martinnet R, Morlet J, Grossmann A. Analysis of sound patterns through wavelet transforms [J]. International Journal of Pattern Recognition and Artificial Intelligence, 2012, 1(2): 273-302.
- [11] 张光新, 崔扬, 周泽魁. 基于小波包分解的纹理图像去噪 [J]. 华南理工大学学报: 自然科学版, 2005, 33(3): 31-33.
- [12] Roerdink J, Meijster A. The watershed transform: definitions, algorithms and parallelization strategies [J]. Fundamenta Informaticae, 2001, 41: 187-228.
- [13] 王宇, 陈殿仁, 沈美丽, 等. 基于形态学梯度重构和标记提取的分水岭图像分割 [J]. 中国图象图形学报, 2008, 13(11): 2176-2180.
- [14] 罗玲, 解梅, 陈杉. 基于多尺度形态滤波的分水岭图像分割方法 [J]. 计算机辅助设计与图形学学报, 2004, 16(2): 168-173.
- [15] 冯丹琴, 盛秋琴, 陈凯, 等. 几种常见滤波器的特性 [J]. 光通信技术, 2002, 26(1): 46-53.
- [16] 李彦丽, 金东瀚, 焦秉立. 几种典型的感知视频质量评价模型 [J]. 计算机工程与应用, 2002, 38(13): 66-68.
- [17] Li G L, Wang N N, Zhang Q. Study of moving image quality evaluation base on MPEG 2 system [J]. Journal of Tianjin University, 2001, 34(5): 573-576.

(上接第 107 页)

- RRT growth [C] // Proc of IROS. [s. l.]: [s. n.], 2003: 1178-1183.
- [3] 徐娜, 陈雄, 孔庆生, 等. 非完整约束下的机器人运动规划算法 [J]. 机器人, 2011, 33(6): 666-672.
- [4] 王滨, 金明河, 谢宗武, 等. 基于启发式的快速扩展随机树路径规划算法 [J]. 机械制造, 2007, 45(12): 1-4.
- [5] LaValle S M. Rapidly-exploring random trees: a new tool for path planning [R]. Ames: Iowa State University, 1998.
- [6] Amin J N, Boskovic J D, Mehra R K. A fast and efficient approach to path planning for unmanned vehicles [C] // Proc of AIAA guidance, navigation, and control conference. Keystone, Colorado: AIAA, 2006: 1-9.
- [7] 樊晓平, 李双艳. 带滚动约束轮移式机器人动态规划的研究 [J]. 控制与决策, 2005, 20(7): 786-788.
- [8] 李磊, 叶涛, 谭民, 等. 移动机器人技术研究现状与

未来 [J]. 机器人, 2002, 24(5): 475-480.

- [9] 宋金泽, 戴斌, 单恩忠, 等. 一种改进的 RRT 路径规划算法 [J]. 电子学报, 2010, 38(2A): 225-228.
- [10] Jouandeau N. Rapidly-exploring sorted random tree: a self adaptive random motion planning algorithm [J]. Lecture Notes in Electrical Engineering, 2009, 24(5): 63-73.
- [11] 席裕庚. 动态不确定环境下广义控制问题的预测控制 [J]. 控制理论与应用, 2007, 17(5): 665-670.
- [12] 赵春霞, 唐振民, 陆建峰, 等. 面向自主车辆的局部路径规划仿真系统 [J]. 南京理工大学学报: 自然科学版, 2002, 26(6): 570-574.
- [13] 张纯刚, 席裕庚. 全局环境未知时基于滚动窗口的机器人路径规划 [J]. 中国科学: E 辑, 2001, 31(1): 51-58.
- [14] 康亮, 赵春霞, 郭剑辉. 未知环境下改进的基于 RRT 算法的移动机器人路径规划 [J]. 模式识别与人工智能, 2009, 22(3): 337-343.