

基于 Voronoi 图改进的 A* 算法在机器人路径规划中的应用

任晓兵 郭 敏

(华中科技大学控制科学与工程系系统所 湖北 武汉 430074)

摘要 :文章描述了一种解决大规划空间下机器人路径规划问题的算法。算法在初步规划中采用 Voronoi 图法描述规划空间,使用 A* 算法搜索出一条初步的可行路径;在二次规划中,按照一定的规则在得到的路径周围重建规划空间。重建后能够有效地缩小规划空间的面积,这样使用算法搜索的实时性得到提升。使用 C++ 软件平台进行仿真的结果显示,多数情况下优化后规划空间面积可以减少一半以上,优化效果明显。

关键词 :机器人 ;路径规划 ;栅格法 ;Voronoi 图 ;A* 算法

中图分类号 :TP242

文献标识码 :A

文章编号 :1009-2374 (2012)21-0033-04

1 概述

机器人的路径规划是机器人研究的一个重要领域,在机器人路径规划领域,很多学者做了大量的研究,研究出多种有效解决问题的办法,在这些方法中栅格法由于其简单、易实现的优点在各个领域得到了广泛的应用。

使用栅格法解决大规划空间的问题时,简单地利用栅格法描述规划空间很难同时满足精度和实时性的要求;由于规划空间较大,在满足一定精度的前提下栅格数量将非常多;而将单个栅格面积设置得较大时,又无法满足精度的要求。很多学者从改变搜索算法的角度来改进基于栅格法的路径规划,如文献采用蚁群算法来搜索栅格空间、文献采用模拟退火算法来搜索栅格空间,但这两种搜索方法在解决较大规模的规划空间时同样会存在组合爆炸的问题,并不能减小栅格的规模。而大多数情况下,栅格空间内障碍物的个数和栅格数一般相差几个数量级。针对这种情况,本文采用基于 Voronoi 图的 A* 算法进行初步规划,在得到的可行路径周围按照一定的规则重建规划空间,重建后能够显著地缩小

规划空间的面积,这样使用栅格法重新描述环境,相同精度下栅格数大幅减少,利用 A* 算法进行搜索时,需要搜索的栅格数目大幅减少。使用软件进行仿真的结果显示,在多数情况下,优化后规划空间面积可以减少一半以上,优化效果明显。这样在相同栅格精度下,规划一条路径所需的时间大幅减少,表现出良好的实时性。

2 路径规划算法的设计和主要实现

该算法由两步规划组成:第一步规划首先生成规划空间障碍物的 Voronoi 图,使用 A* 算法规划出初步的路径;第二步规划则在第一步生成的可行路径的基础上重构规划空间,使用栅格法描述重构后的规划空间,并再次使用 A* 算法进行路径搜索,得到最终的精度较高的可行路径。

2.1 构建障碍物的 Voronoi 图

Voronoi 图是一种关于空间划分的基础数据结构,在求解点集或其他对象的与距离相关的对象时有重要的作用。它将平面划分成凸多边形形状的 Voronoi 区域,使得区域内的任何点距离当前站点比距离其他站点近。

在规划环境中,机器人的外形各异,并且障碍物的形状、大小也各不相同,为了简化问题,要对规划空间做一些预处理。首先,因为机器人是在二维环境下,可以不考虑其高度信息;其次,因为机器人是有长度和宽度的,当两个障碍物靠得太近,即两个障碍物最近距离小于机器人通过的安全距离时,要将两个障碍物合并成一个障碍物,这样机器人就可以在预处理后的任何两个障碍物之间通过;最后做障碍物的外接圆,并进行径向扩张。经过这样处理,机器人可简化成质点处理,障碍物可以用其外接圆的圆心表示,问题就变成了点机器人在平面规划环境下的路径规划,而不同形状的障碍物可以用质点表示。我们以规划环境的左下角作为坐标原点,横向向右为X轴正方向,纵向向上为Y轴正方向,建立直角坐标系。

经过上面的环境预处理,提取出了生成Voronoi图的生成元,接着可以利用这些Voronoi元即障碍物外接圆的圆心来生成障碍物的Voronoi图。生成Voronoi图的方法有很多,我们这里采用逐点插入法,利用逐点插入法对描述障碍物的点进行Delaunay三角划分,由于Voronoi图是Delaunay图的偶图,经过对偶处理可得到障碍物的Voronoi图。

2.2 基于voronoi图使用A*算法搜索最短路径

A*算法是在Dijkstra算法的基础上发展起来的算法,从起始节点出发,依次对当前节点的子节点权重进行更新,并用子节点中权重最小者对当前节点更新,直至遍历所有节点为止。A*算法考虑了移动机器人目标点位置信息,沿着目标点位置开始搜寻,与Dijkstra算法相比路径搜寻效率更高。

在前面得到的描述环境的Voronoi图的基础上,对于已经建立的节点网络,在考虑环境边界的情况下,我们重新定义图中的网路节点,对于在环境边界上的节点我们使用该节点所在边的中点来代替该节点,对于不在边界上的节点则保持不变,在按规则生成的新的路径网络中使用A*算法得到最短规划路径。在A*算法中我们做以下设定:

S——存放路径网络中所有节点的集合;

S_i ——路径规划搜索过程中的路径网络中某一

个节点;

x_i, y_i ——分别为网络节点 S_i 的横、纵坐标;

$g(i)$ ——初始节点 S_0 到 S_i 的估计距离;

$h(i)$ ——计算 S_i 到终点E的启发距离;

$f(i) = g(i) + h(i)$, 节点 S_i 的距离评价函数;

Open——存放等待扩展的节点的队列集合;

Close——存放已经扩展过的节点的队列集合;

F_e ——扩展节点的函数,扩展当前节点的邻接节点;

F_i ——插入节点函数,判断当前节点的邻接节点是否要插入到Open表,并更新其估计值;

D_i ——计算节点 S_i 评价值的函数,在算法中采用欧式距离作为评价距离。

在这里采用两个节点之间的欧式距离作为评价函数,距离可由两个节点的坐标得到。两个节点 S_i 和 S_j 的距离如下公式所示:

$$D(i, j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

其中 x_i, y_i 为节点 S_i 的横、纵坐标, x_j, y_j 为节点 S_j 的横、纵坐标。A*算法流程如下:

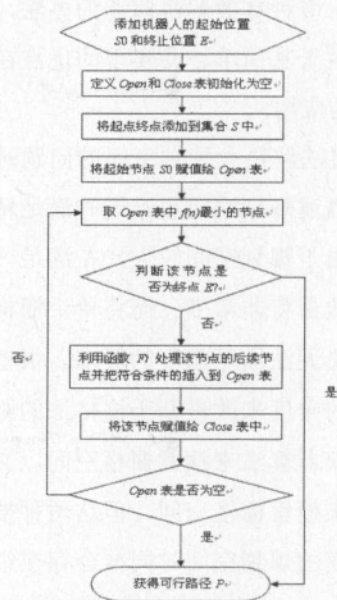


图1 A*算法流程图

在该算法中利用函数 F_e 来得到当前节点的邻接节点,利用函数 F_i 处理当前节点的邻接节点,在生成Voronoi图的过程中,可以得到一个记录节点的

邻接节点索引的集合, 函数Fe就要是利用该集合来得到当前节点的邻接节点的索引的。在算法中利用Di来计算评价函数 $f(i)$ 的值。Fi函数是处理当前节点的邻接节点的算法, 当节点不在Open和Close表中时, 计算其估计值排序后插入到Open表中; 当在Open表中时, 比较节点新的估计值Di和Open表中的估计值, 选择其中小的作为其新的估计值。如果节点在Close表中, 比较节点新的估计值Di和Close表中的估计值, 如果Di小于Close表中的估计值时, 将该节点从Close表中移除, 添加到Open表中。而得到的路径节点存放在Close表中。搜索得到的路径。

2.3 重构规划环境进行二次路径规划

根据初步规划得到的可行路径, 在求得的可行路径节点序列中(包括起点和终点), 在得到的这些节点中, 寻找一组最大和最小的纵横坐标, 前面计算中我们都是将机器人当做质点考虑的, 考虑到机器人的外形, 可以对得到的坐标进行处理, 这里用长方形作为机器人的外形, 在最大的纵横坐标上分别加上机器人外接长方形的长、宽, 最小的纵横坐标分别减去机器人的外接长方形的长和宽, 便可得到一组最终的坐标 X_{MAX} , X_{MIN} , Y_{MAX} , Y_{MIN} 。

这里设规划空间的长为 $Length$ (X轴方向), 宽为 $Width$ (Y轴方向), 然后按照不同的情况, 做以下的处理:

(1) 当 $X_{MAX} - X_{MIN} < 1/2 \times Length$ 或 $Y_{MAX} - Y_{MIN} < 1/2 \times Width$ 至少有一个成立时, 以 (X_{MIN}, Y_{MIN}) , (X_{MIN}, Y_{MAX}) , (X_{MAX}, Y_{MIN}) , (X_{MAX}, Y_{MAX}) 为顶点确定的长方形作为新的规划空间。

(2) 当 $X_{MAX} - X_{MIN} < 1/2 \times Length$ 和 $Y_{MAX} - Y_{MIN} < 1/2 \times Width$ 都不成立时, 即最大最小坐标的差值比较接近规划空间的长和宽时, 做以 $(Y_{MAX} - Y_{MIN}) / (X_{MAX} - X_{MIN})$ 为斜率的两条直线, 使所有路径节点都在两条直线的不同两侧, 且使所有节点到直线的距离都为一个定值。这个定值至少要大于其外接长方形的长和宽。然后求得两条直线和规划环境的边界的交点 X_1 , X_2 , Y_1 , Y_2 , 接着求包括起点和终点在内的6个点确定的凸壳, 这个凸壳就是我们确定的新的规划空间。

由于在初次规划中, 在新的规划空间中存在一条可行路径, 而且是最优或者次优路径, 使用栅格法在新的规划空间中也一定能够找到一条可行路径。使用这种优化方法, 二维空间下, 规划空间面积大幅缩小, 规划空间中的障碍物的个数也相应地减少了。经过处理后, 问题重新归结为基于栅格环境的机器人路径规划时, 栅格环境下使用A*算法的路径规划有着比较成熟的方法, 这里不再重点描述。

3 仿真实验和结果分析

采用的实验环境是: CPU Intel Core2 Duo, 内存2GB, C++编程工具进行仿真, 初始的规划环境是以左下角为坐标原点, 坐标为 500×500 的规划区域, 障碍物随机分布, 经过预处理, 构建障碍物的Voronoi图, 在图上设定好起点和目标点坐标后, 使用A*算法进行路径搜索, 得到初步的可行路径, 在得到的可行路径的基础上使用本文描述的重构规则重构规划空间, 得到的结果如下图所示:

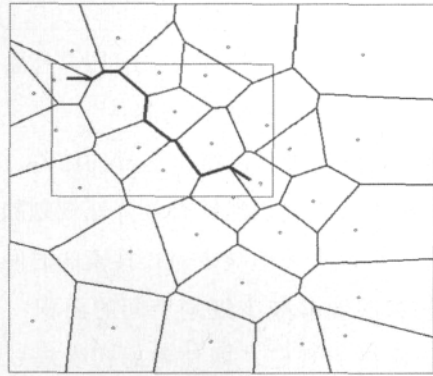


图2 根据规则1形成的新的规划空间

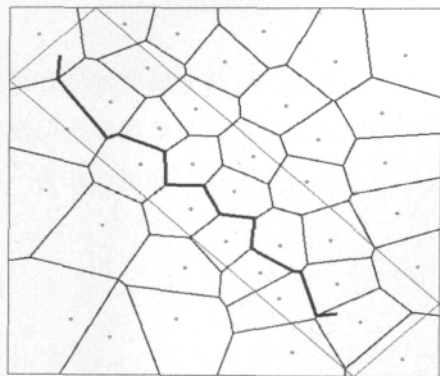


图3 根据规则2形成的新的规划空间

如图2所示是按照规则1重构的规划空间，初步规划时的规划面积约为重构后的规划面积的5.4倍。如图3所示是按照规则2重构的规划空间，初步规划时的规划面积约为重构后的规划面积的3.1倍。可以看到两种情况下，重构后的规划空间和初步规划的规划空间相比规划面积减少一半以上，能够有效地缩减规划空间的面积。

接着，我们分别在 100×100 、 500×500 、 1000×1000 的栅格规模下进行仿真试验，试验条件如下：每种栅格环境下使用A*算法进行10次路径规划；进行随机障碍物，随机起点和目标点的路径规划；规定障碍物的个数是100个。

则对每种栅格规模进行10次路径规划，求其平均规划时间，如下表所示：

表1 不同栅格规模下规划一条路径的平均时间对比表

栅格规模	规划一条路径的平均时间（秒）
100×100	0.74
500×500	6.34
1000×1000	34.54

而得到100个障碍物的Voronoi图并搜索出一条可行路径的平均时间是约为0.64s。

结合实验的结果可以得出这样的结论，在较大的栅格规模下，当我们能够有效降低规划的栅格规模，这样就可以有效地减少路径搜索的时间，而使用该优化算法能够减少规划空间的面积，实际就是降低了栅格的规模。即使考虑初步规划中使用Voronoi图进行路径规划的时间，其实时性仍然可以得到明显的改善。

4 结语

机器人路径规划在机器人研究领域有重要的意义，但目前的路径规划方法虽然很多，但没有哪一种方法是适用于所有情况的。栅格法作为一种被普

遍应用的方法，使用Voronoi图来进行改进是有一定的现实意义的。仿真的结果证明该方法可以有效地减少规划空间的面积，能够明显地减少栅格的数量，降低了栅格的规模，加快了路径规划算法的搜索速度，提高了路径规划的实时性，是一种可行的优化思路。但本文还存在不足的地方，如在得到初步规划的规划路径后怎样设计出更有效和更合理的规则来减少规划空间的面积，还需要进一步的探讨和研究。②

参考文献

- [1] Y.Ma ,M.Zamirian . A novel approach for multiple mobile objects path planning parametrization method and conflict resolution strategy[J] . Physics Letters A 2012 376 (4) .
- [2] 朱庆保 张玉兰 . 基于栅格法的机器人路径规划蚁群算法 [J] . 机器人 2005 27 (2) .
- [3] 郑秀敏 顾大鹏 刘相术 . 基于栅格法 - 模拟退火法的机器人路径规划 [J] . 机器人技术 2007 23 (2) .
- [5] 李圣权 胡鹏 杨传勇 . 图形部件 Voronoi 图生成算法与应用研究 [J] . 计算机工程 2005 31 (10) .
- [6] 许松清 吴海彬 . 基于 Voronoi 图法的移动机器人路径规划 [J] . 中国工程机械学报 2005 3 (3) .
- [7] Alexopoulos C ,Griffin P M . Path planning for a mobile robot[J] . IEEE Transactions on System Man and Cybernetics 1992 22 (2) .
- [8] 马勇 . 基于平滑 A* 的移动机器人路径规划 [J] . 同济大学学报(自然科学版) 2010 38 (11) .

作者简介：任晓兵(1985 -)男 华中科技大学控制科学与工程系系统所硕士研究生 研究方向 机器人路径规划。

(责任编辑 周加转)