

基于层次地图的子目标点的选择与更新策略

刘宏, 王健, 王闯奇

(北京大学 信息工程学院, 深圳 518055)

摘要: 为了适应较大场景下多自由度移动机器人的路径规划, 建立了一种新型的分层采样地图. 利用第一层二维地图进行子目标点的选择和更新, 从而实现子目标点对路径规划的全局引导; 然后结合局部规划器和第二层九维地图进行路径搜索得到路径. 考虑当前局部空间的子目标点存在被运动障碍物遮挡的情况, 若当前子目标点被遮挡, 则选取下一个子目标点为当前子目标点进行规划; 若不能找到有效子目标点, 则反馈到第一层进行子目标点更新, 更新后再继续进行路径搜索. 通过这种上层引导、下层更新反馈的机制对 PRM(probabilistic roadmap method)进行改进, 在保证概率完备性的同时, 提高了路径搜索的速度. 文中定义了子目标点区域对两层地图之间的结合进行了优化, 并引入双向 RRT 进行局部地图增强以解决困难区域问题. 仿真实验表明该方法能达到很好的实时性和鲁棒性.

关键词: 路径搜索; 分层采样; 子目标点; 局部规划

中图分类号: (作者本人填写)

文献标识码: A

文章编号: IR11-085

Subgoal Selection and Updating Strategy based on Hierarchy Sampling Strategy

Hong Liu, Jian Wang, Chuangqi Wang

(School of information engineering, Peking University, Shenzhen, 518055, China)

Abstract: A new kind of subgoal selection and updating method was presented in the probabilistic roadmap which was used to tackle the high dimensional robot path planning in the large scene. In order to execute the method in low computation complexity, multi-level hierarchy sampling strategy(HSS) was altered and introduced and subgoal processing was dealt with in the first level roadmap. The second level roadmap was used to search the local path with local planner. One important matter being faced with during the path planning phase was that the current subgoal becoming invalid results from the moving obstacles. In our method, when the current subgoal become invalid, the next subgoal was chosen to replace the current subgoal. If still here is no a valid one, subgoals between the current subgoal and goal were all updated in the first level roadmap. Subgoal area was defined to optimize the connection between the first level and the second level map. Local RRT was used to enhance the local map to solve the difficult areas planning. Simulation results show that by this method real-time and robustness can be achieved.

Key words: path planning; hierarchy sampling strategy; subgoal; local planner

鉴于路径规划经典算法在高自由度机器人领域的局限性, 90 年代提出了位姿空间^[1]的概念. 基于位姿空间的路径规划根据环境的不同可以分为静态路径规划和动态路径规划. 研究者提出了多种经典算法用于解决静态路径规划问题, 如 PRM (probabilistic roadmap method)^[2]、Lazy PRM^[3]、RRT (rapidly exploring random tree)^[4]等.

尽管这些方法可以解决大部分静态环境下的路径规划问题, 但现实场景并不总是静止不变的, 因此这些方法处理动态问题有局限. 动态规划引

入的难点问题包括: 动态障碍物增多带来的路径失效率增大问题; 环境的增大带来的计算复杂度升高问题. 经典的方法如 DRM^{[5][6]}和 DRRT^[7]被提出用于解决简单的动态问题, 但当环境过于复杂或者变化非常频繁的时候, 计算复杂度非常大, 不能满足实时性要求. 因此本文提出了一种利用较少信息进行全局引导, 同时进行局部规划的方法.

1 路径规划算法框架

本文要解决带有移动底座的机械臂的路径规划问题. 我们使用机器人前两个自由度, 即位置信息来启发引导机器人路径规划. 提出了一个两层的 PRM 方法, 同时有上层引导下层规划局部路径, 并

收稿日期: 2011-5-22

作者简介: 刘宏, 男, 教授; 深圳, 北京大学, 信息工程学院, 518055. E-mail: liuh@szpku.edu.cn

基金项目 国家自然科学基金(No.60875050), 广东省自然科学基金(NO.9151806001000025), 深圳市科技计划及基础研究项目(JC200903160369A)

由下层到上层反馈的机制. 图 1 为本文方法的框架图. 该方法建立两层地图, 分别为引导层和寻路层. 在引导层生成机器人行走的大致方向信息, 然后将引导信息作用在寻路层来实现细致的路径搜索, 当路径执行时如果碰到障碍物, 则首先启用局部规划器得到一条局部路径, 如果规划局部路径失败, 则反馈给引导层, 通过对引导信息的调整, 使路径搜索可以继续下去, 最终到达目标位置. 由于子目标点易获得且易更新, 因此本文用子目标点来储存引导信息, 信息在两层地图之间的反馈也是通过子目标点更新来传递.

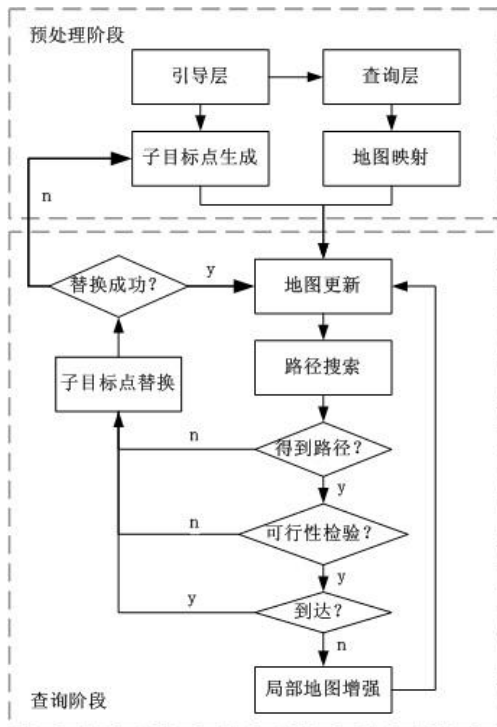


图 1 路径规划整体框架图

2 分层地图的搭建

由于场景较大, 机械臂的维度也较高, 为了更有效的对路径进行规划, 需要建立一个足够健壮的 Roadmap, 在此采用多层 Roadmap. 建立多层概率地图的传统方法为 HSS^[8]. 如图 2 (a) 所示.

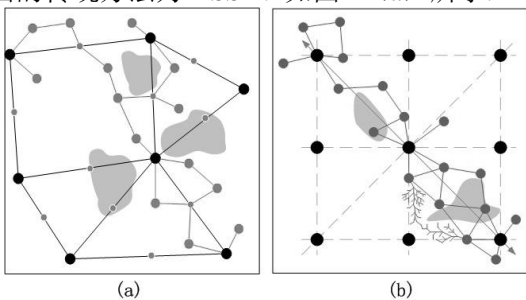


图 2 (a) HSS 三层地图示意图 (b) 双层地图采样示意图

但对移动机器人来讲, 困难区域是动态变化的. 因此 HSS 分层方法存在如下的问题: 通过取中点构成的第二层 Roadmap 的质量, 很依赖第一层地图的质量, 尤其是在第一层采样点较少的情况下, 第二层采样点的质量也随之降低, 此时其概率完备性不及相同采样数目下的平均采样; 第三层地图用于离线解决动态困难区域问题, 灵活性较差. 在此提出了改进的分层地图构造策略.

在改进的地图构造策略中, 取消了 HSS 方法中由中点构成的第二层采样点. 第一层结点只对底座 2 个维度进行采样, 因此采用完备性较好的栅格法^[9]可以很快的完成, 其它 7 个维度值置为固定值. 结点生成后无需进行 $K_{nearest}$ 策略连接, 只需采用 4 邻域或者 8 邻域连接策略即可 (本文实验采用 8 邻域), 该层地图用于进行路径估计 (即获得子目标点, 见第 3 节); 第二层结点是在第一层结点的基础上再随机采样, 从而对整体地图进行细化, 采样时对机械臂 9 个维度进行采样, 该层地图用于局部空间的路径规划; 在第二层地图中规划时, 若出现路径规划或执行失败的情况, 则通过地图增强或局部路径在线搜索等方法对当前的局部地图进行在线增强, 然后再继续路径的搜索. 改进的地图如图 2 (b) 所示.

3 子目标点生成与更新

子目标点策略在很多计算复杂和规模较大的路径规划中被引入^{[10][11]}, 本文也采用该策略. 但 these 方法所用到的子目标点都是预先得到的子目标点, 当障碍物发生移动后, 子目标点便会失效. 因此引入子目标点后需要解决的问题主要包括子目标点的生成和子目标点的更新.

3.1 子目标点生成

子目标点生成器由第一层采样点和 A* 算法构成, 通过 $K_{nearest}$ 策略将初始点和目标点添加到第一层采样点, 然后利用 A* 算法得到一条粗略路径, 该路径上的点便当做初始化的子目标点来使用, 如图 3 所示.

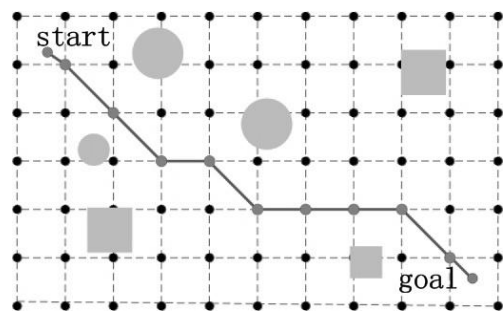


图 3 子目标点初始化示意图

由于寻找子目标点的目的是为接下来的局部路径规划提供一个引导信息，又因为在动态环境下，寻找到的子目标点仍会失效，因此为节省计算资源，在此只进行点检测，不进行边检测。

3.2 子目标点更新

在寻找路径的过程中，由于环境一直在动态变化，所以会出现当前局部规划用到的子目标点被障碍物遮挡失效的情况。这时，需要进行子目标点的更新。子目标点更新有两种方式：

a. 局部更新. 选取当前子目标点的下一个子目标点作为当前子目标点，并将当前路径片段上最新的有效点作为局部初始点。然后利用 WC-Mapping 进行局部的 A* 搜索或者用双向 RRT 方法^[12]对该局部区域内进行路径搜索。如图 4 所示，由于当前子目标点 B 被障碍物遮挡，因此调用下一个子目标点 C，由当前 O 点向 C 点规划一条路径，通过 O 和 C 双向生长 RRT 树，最终得到一条局部路径；

b. 全局更新. 通过子目标点生成器重新生成包括当前子目标点在内之后的所有子目标点。该方法需要在引导层对粗略路径进行全局更新。如图 5 所示，A 点之后的 B、C、D 三点都被障碍物遮挡，此时采用局部更新失效，需要在引导层对 A 点之后的子目标点进行更新。更新时需考虑当前运动障碍物的位姿空间占据情况。

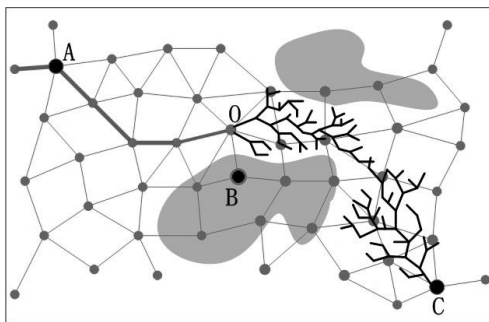


图 4 子目标点局部更新及局部 RRT 路径搜索

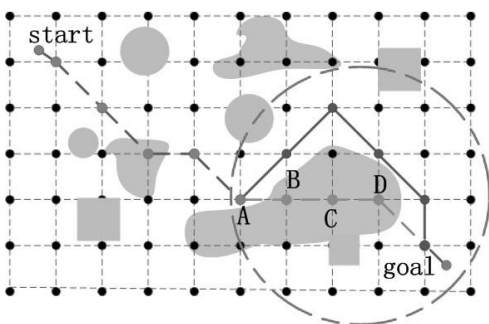


图 5 子目标点全局更新

由于场景较大，多次的全局更新很可能导致更大的计算耗散，因此当出现寻找路径失败时，先采取局部更新的方法，当局部更新失效，再采用全局更新。实验中利用机械臂的底座二维信息定义了子目标点区域，一旦机械臂进入该区域即作为到达子目标点处理。

4 实验及分析

实验所用的 Kawasaki FS03N 型机械臂如图 7 所示。该机械臂具有 6 个自由度，移动底座有 3 个自由度，因此整个移动机械臂的自由度是 9。同时，本文规划器使用 C++ 实现并使用开源的三维碰撞检测包 ColDet1.2 来进行碰撞检测。实验结果均是在处理器 2.31GHz，内存为 3GB 的 PC 下得出。下面对实验场景进行描述并对实验结果进行分析。

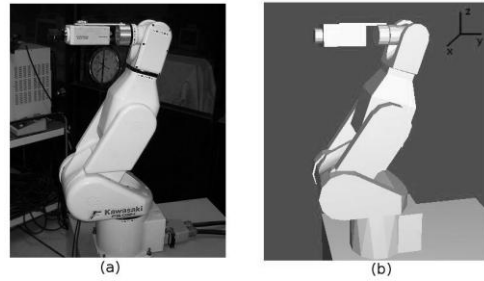


图 7 Kawasaki FS03N 机械臂的仿真

4.1 实验场景

本文设置实验场景如图 8 所示。环境中障碍物分为两类：第一类为悬挂式，记为 O_1 ；放置在地面的障碍物记为 O_2 ，两类障碍物仅可一维平移运动。其中，两实验场景的尺寸均为 $800 \times 600 \times 100$ 。场景中仅包含 O_1 类型的障碍物，个数为 14，并且每 7 个排成一排，共 2 排。

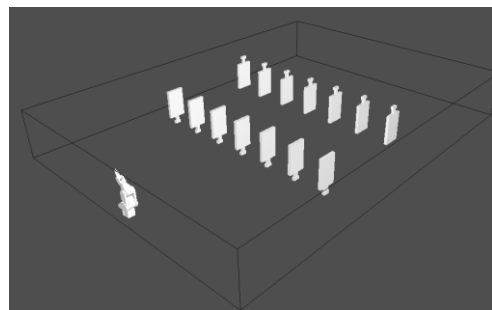


图 8 实验场景

4.2 实验结果和分析

我们的方法是对 DRM 算法的一种改进，即加

入了引导层,从而使搜索局部化.与 DRM 相比,本方法能更好的启发路径搜索,尤其是对重规划问题的解决上有很大的优势. DRM 则不能很好的应对重规划问题.

表 1 显示了我们的方法在场景中的多次实验的结果,其中总采样点数目为 4000,第一层采样点数目是渐变的,表中 *Node.num* 表示第一层采样点的近似数目值. *Grid Size* 表示对应于 *Node.num* 的近似方格尺寸. *T. time* 表示完成一次搜索任务的总时间; *Succ. rate* 表示成功率. 实验结果显示,当第一层采样节点增加到一定程度后,成功率下降.这是由于子目标节点足够接近,造成子目标点冗余,由于频繁的进行子目标点的更新,导致总时间会有所提高且成功率下降. 因此,存在合理的第一层节点个数. 由表 2 可知,在该实验场景中,第一层节点的个数为 400 左右时比较合理.

表 1 场景中的实验结果

<i>Task</i>	<i>Node.num</i>	<i>Grid-Size</i>	<i>T time/s</i>	<i>Succ. rate</i>
001~050	100	60×60	10.11	66%
051~100	300	40×40	20.87	70%
101~150	400	30×30	19.54	74%
150~200	1000	20×20	31.89	46%

5 结 论

本文提出了一个新的启发式的两层路径规划方法. 通过前两个自由度对机器人进行路径引导,确定子目标点. 然后在子目标点之间使用 DRM 进行精确路径规划. 若路径失效,首先采用局部增强策略对局部区域增强并试图找到一条短路径. 如果寻找短路径失效,则反馈给上层,更新上层子目标点. 进而得到新子目标点. 如此迭代直到到达目标点. 仿真结果表明,在动态环境中,该方法能较好的完成规划任务,并且在路径更新、优化、实时性上有明显的优势. 本文仅仅对自由度分治进行了初步的探索,在未来的工作中,可以加入自适应策略使该方法更有效. 总之,虽然该方法仍有需要完善的地方,但对于大尺度的动态环境,本文提出的算法是一个很有效的方法.

参考文献:

[1] S.M. LaValle. Planning Algorithms[M]. Cambridge, U.K.: Cambridge University Press, 2006.
 [2] L.E. Kavraki, P. Svestka, J.C. Latombe, and M. H. Overmars. Probabilistic roadmaps for path planning in high-dimensional configuration spaces[J]. IEEE

Transaction on Robotics and Automation, 1996, 12:566-580.
 [3] R.Bohlin and L.E.Kavraki. Path planning using Lazy PRM[C]//IEEE International Conference on Robotics and Automation. San Francisco, CA, 2000:521-528.
 [4] S. M. LaValle. Rapidly-exploring Random Tree: a new tool for path planning[C]//Technical Report: No.98-11. Iowa State University, Dept. of Computer Science, 1998.
 [5] P. Leven and S. Hutchinson. Toward real-time path planning in changing environments[C]//Proceedings of the Fourth International Workshop on the Algorithmic Foundations of Robotics, 2000: 363-376.
 [6] P. Leven and S. Hutchinson. A framework for real-time path planning in changing environments[J]. The International Journal of Robotics Research, 2002, 21:999-1030.
 [7] D. Ferguson, N. Kalra and A. Stentz. Replanning with RRTs[C]//IEEE International Conference on Robotics and Automation. 2006: 1243-1248.,
 [8] D. Ding, H. Liu, X. Deng. A dynamic Bridge Builder to identify difficult regions for path planning in changing environments[C]//IEEE/RSJ International Conference on Intelligent Robots and Systems. 2007: 2925 - 2931.
 [9] R. Geraerts, M.H. Overmars. A comparative study of probabilistic roadmap planner[C]//Proc. of the Fifth Int. Workshop on the Algorithmic Foundations of Robot. 2002: 249-264.
 [10] P. C. Chen and Y. K. Hwang. SANDROS: A dynamic graph search algorithm for motion planning[J]. IEEE Transactions on Robotics and Automation, 1998: 390-403.
 [11] S. Candido, Y. T. Kim and S. Hutchinson. An improved hierarchical motion planner for humanoid robots[C]//IEEE-RAS International Conference on Humanoid Robots. 2008: 654-661.
 [12] J. James, Jr. Kuffner, S. M. LaValle. RRT-connect: An efficient approach to single-query path planning[C]//IEEE International Conference on Robotics and Automation. 2000, 2: 995-1001.