

基于区域分类的机器人动态环境安全路径规划

丁润伟¹ 马佳瑶² 康日升¹ 钟铖铖¹

(1 北京大学深圳研究生院深圳物联网智能感知技术工程实验室, 广东 深圳 518055;

2 中国农业大学信息与电气工程学院, 北京 100083)

摘要 提出了一种基于区域分类的安全路径规划方法,通过计算区域拥塞程度和稳定程度,做出安全性评判,选择适合的方法进行扩展,同时在重规划的采样过程中,通过概率自适应的方法选择导航点,达到平衡 RRT 树的效果.9 自由度的机械臂仿真实验表明:该方法对于动态环境下的安全运动规划问题有较好的效果.与 MPRRT 比较,本文方法完成任务的平均时间加快 31.60%,重规划次数降低 18.37%,规划成功率提高 2.5%.

关键词 路径规划; 动态环境; 区域分类; 概率自适应; 快速搜索随机树法

中图分类号 TP242 **文献标志码** A **文章编号** 1671-4512(2015)S1-0312-03

Region classification based robot security path planning in dynamic environment

Ding Runwei¹ Ma Jiayao² Kang Risheng¹ Zhong Chengcheng¹

(1 Engineering Lab on Intelligent Perception for Internet of Things(ELIP),

Shenzhen Graduate School, Peking University, Shenzhen 518055, Guangdong China; 2 College of

Information and Electronic Engineering, China Agricultural University, Beijing 100083, China)

Abstract Variant of rapidly exploring random tree (RRT) was proposed which was called region guided RRT (RGRRT). It aims to find a path in changing environments efficiently by considering regional congestion and regional stability and choose different method to extend. During replanning phase, RRT selected the waypoints by probabilistic adaptive approach. This method focused on security considerations during replanning. RGRRT possess probabilistic completeness guarantee as well as improving the speed of searching path. High degrees of freedom manipulators simulation experiments' results show that RGRRT improves efficiency significantly for the safe path planning. Compared with MPRRT, RGRRT speeds up the average time 31.60%, reduce the replanning times 18.37%, improve the success rate of 2.5%.

Key words safe path planning; dynamic environment; region classification; rapidly exploring random tree

路径规划算法的任务是得到一个动作命令的序列,规划对象通过执行该动作命令序列可以从初始状态运动到目标状态,并且在运动过程中不发生任何类型的碰撞(包括自碰撞以及与障碍物的碰撞).经典的路径规划方法有路径地图法^[1]、胞腔分解法^[2]、势场法等^[3].但如果位姿空间维数大于 3 维,这些方法就几乎无法使用了.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 年代提出了基于采样的规划方法,其中包括两种经典的框架:概率地图 PRM (probabilistic roadmap method)^[4]和快速搜索随机树法,如 RRT (rapidly exploring random tree)^[5];基于这两种框架的多种算法,如动态路图法 (DRM)^[6]、动态搭桥法 (DBB)^[7]、动态快速搜索随机树法

收稿日期 2015-06-30.

作者简介 丁润伟(1983-),女,工程师, E-mail: dingrunwei@pku.sz.edu.cn.

DRRT^[8]、随时动态快速扩展树法 ADRRT^[9] 和多分支快速扩展树方法 MPRRT^[10] 等。但是, 这些方法很少关注障碍物运动情况的信息, 导致运动场景较为困难的情况下, 如障碍物比较密或者障碍物运动较为频繁等, 花费大量时间计算出的路径很容易失效。

本研究提出了一种基于区域分类的安全路径规划方法。通过节点区域稳定性和拥塞程度, 进行不同方式的扩展, 规划一条时间持续性较长的路径, 减少重规划次数, 同时在重规划的采样过程中, 通过概率自适应的方法选择导航点, 达到平衡 RRT 树的效果。

1 算法概述

算法的流程图如图 1 所示。

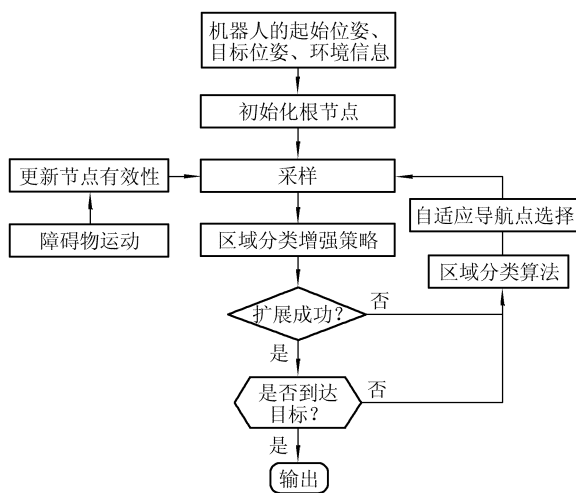


图 1 基于困难程度与稳定性的区域划分方法流程图

算法流程以三部分为重点: a. 区域分类方法, 在采样过程中, 记录节点的区域信息并进行分类; b. 根据分类结果, 对于节点采取不同的扩展策略; c. 在重规划过程中, 采取概率自适应策略使得导航点在目标点和安全区域之内可以更多地采样。

2 基于困难程度与稳定性的区域分类方法

传统的区域分类方法对于边界非常敏感, 计算量巨大, 针对此不足, 本文算法采用观测窗口的方法进行区域分类。

2.1 拥塞程度分析

在窗估计方法中, 某一观察点的概率密度函数可以用其周围节点来估计。整个窗口内的概率密度总和可以定义为

$$P(x) = \frac{1}{n} \sum_{i=0}^n \frac{1}{h^d} \binom{n}{k} K\left(\frac{x-x_i}{h}\right), \quad (1)$$

式中: x_i 为窗口中的随机观测点; h 为窗口大小; K 为 d 维中的窗口函数。由此类推, 本文中的观测信息其概率总和可以定义为

$$P(q) = \sum_w q_{inv} / [\sum_w (q_{inv} + q_v)], \quad (2)$$

式中: q 为窗口的中心点; q_v 为该点有效; q_{inv} 为该点无效; w 为观测窗口。窗口内无效点 q_{inv} 比例越高, 窗口内区域环境越拥塞。

考虑到环境是连续变化的, 可以通过观察父节点的拥塞程度来平滑当前节点, 即

$$f = \omega e(q_{new}) + (1 - \omega) e(q_{nearest}), \quad (3)$$

根据环境的变化修改 ω 值以优化算法性能。

2.2 稳定性分析

分析稳定性时, 须开设时间窗口, 统计区域内节点有效性变化的频率。令 $P+$ 代表点从原本阻塞状态变为自由状态, $P-$ 代表点从自由状态变为阻塞状态。则有效性变化点 M_c 为 $P+$ 和 $P-$ 点的总和。有效性不变化的点总和为 M_s 。以中心节点 CentralNode 为圆心固定半径 r 的窗口, 以为时间观测的窗口, 可以得到观测节点区域 w 在时间内的有效性变化情况,

$$P(q) = \sum_w M_c / [\sum_w (M_s + M_c)], \quad (4)$$

3 基于区域分类的节点增强策略

在节点增强阶段, 算法根据区域稳定程度和拥塞程度情况进行扩展, 分为 S_1, S_2, S_3 三类, 分别采用原始、水平、垂直方法进行增强。

稳定性较低的节点组成 S_1 区域, 这些节点增强后容易失效, 考虑到算法的实时性, 选择时间复杂度最低的原始 RRT 扩展方法; 稳定性较高, 拥塞程度适中的节点组成 S_2 区域, 这种区域最典型的代表是自由位姿空间和障碍物的交界区域, 采取时间复杂度适中的垂直方向扩展; 稳定性较高, 拥塞程度又比较大的点, 构成 S_3 区域, 为了保证规划成功率, 在前两种方法失败的情况下, 采取时间复杂度较高的垂直方向扩展。

原始 RRT 扩展方法时间复杂度最低, 垂直方向扩展方法时间复杂度适中; 水平扩展方法时间复杂度最高, 但是可以一定程度上保证此点扩展的成功, 其在 $\overrightarrow{q_{nearest} q_{rad}}$ 上通过二分法找到边界点, 然后生成一个小的随机偏移角, 在射线上找到另一个边界点 q_2 , 沿 $\overrightarrow{q_1 q_2}$ 方向扩展。如图 2 所示, 浅色节点组成 S_1 区域, 灰色节点组成 S_2 区域, 深色节点组成 S_3 区域。

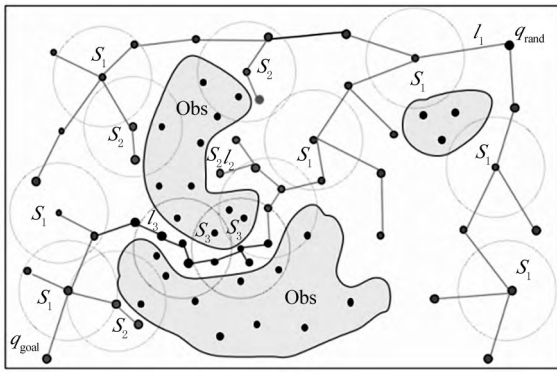


图 2 基于区域分类的节点增强方式

4 概率自适应导航点选择策略

在重规划中,结合环境变化的特点,可以将导航点分为三个种类:随机点 q_{rand} , 目标点 q_{goal} , 裁剪下来的有效子树上的有效点集 f .

算法根据导航点扩展是否成功,来调节导航点选择概率 P_N 的大小. 方式如下:若扩展成功,则以 A/α 的倍数增大对应类别选择概率 P_N 的大小;若扩展失败,则以 $A\alpha$ 的倍数减小对应的 P_N .

对比固定的经验值划定导航点选择概率的反馈,自适应导航点策略能够更好地平衡、调节树的生长方向.

5 实验及结果分析

为了验证本文算法性能,针对动态环境下安全运动规划问题,试验设计了仿真机器人,其为 9 自由度的 Kawasaki FS03N 型机械臂,如图 3 所示. 实验运行环境为 Intel Core i5, 内存 8 GB, OpenGLFreeglut 2. 8. 1, 碰撞检测包 ColDet1. 2. 仿真平台由 Microsoft Visual Studio 2010 编写.

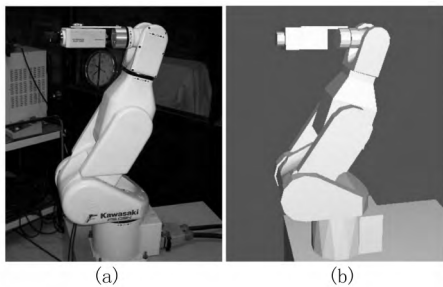


图 3 机械臂及其仿真模型

5.1 实验场景

仿真平台中实验场景大小设计为 $300\text{ cm} \times 300\text{ cm} \times 90\text{ cm}$, 机器人占据的空间是 $30\text{ cm} \times 20\text{ cm} \times 90\text{ cm}$, 每个障碍物的大小是 $10\text{ cm} \times 50\text{ cm} \times 10\text{ cm}$.

场景中障碍物的初始位置随机产生,机器人要完成的任务是从环境右下角移动到到对角线左上角. 为了给机械臂不同的自由度设置障碍,障碍物是可以进行旋转和平移的漂浮障碍物. 其运动由 6 个参数来确定. 每次平移的距离范围是 $\pm 3\text{ cm}$, 旋转角度范围是 $\pm 30^\circ$.

5.2 实验结果和分析

将本文方法与 MPRRT 进行了性能比较,每个场景中进行 50 组实验.

表 1 场景中实验结果

实验参数	MPRRT	RGRRT
平均完成任务时间/s	19. 81	13. 55
平均重规划次数	2. 34	1. 91
最大重规划次数	5	3
成功率/%	86. 00	88. 00

从表 1 可以看出:与 MPRRT 相比,本文算法完成任务的平均时间明显加快了 31. 60%,重规划次数降低 18. 37%,规划成功率提高 2. 5%. 这是因为在本文算法的规划方法中,重点关注增强区域的稳定性,这对于路径的持续有效性有很大的提高;同时,在重规划进行过程中,包含大量启发信息的采样策略使得扩展的成功率也有所提高,导致总成功率略有提高. 综上,本文算法提出的区域分类安全路径规划方法,在动态环境中比较有优秀的性能.

图 4 显示:在实验中同一序号的场景,障碍物个数和运动速度频率相同,随着场景序号增加,障碍物个数、运动速度和频率逐渐增加. 可以看出:在场景较为困难的情况下,本文方法相比 MPRRT 方法能够显著减少重规划次数.

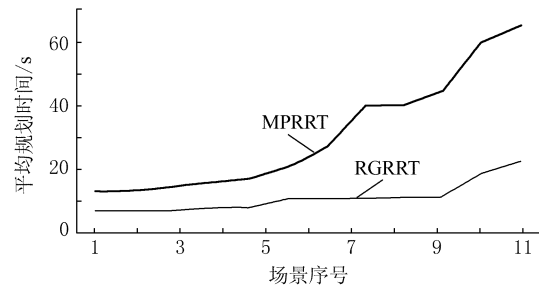


图 4 场景困难程度增加过程中 RGRRT 和 MPRRT 的平均规划时间对比

实验证明,与 MPRRT 方法相比,本文提出方法规划成功率提高 2. 5%,完成任务的平均时间降低 31. 60%,尤其在障碍物运动较为频繁的情况下,能显著减少重规划次数. 未来可以讨论区域大小时间窗口长短选取和不同场景困难程度的关系,并尝试解决场景困难程度的量化描述方式.

(下转第 350 页)

- feedback controller for the path-following and the point-stabilization problems of a unicycle-like mobile robot[C]//Proc of Control Applications, 1997, Proceedings of the 1997 IEEE International Conference on. Hartford, CT: IEEE, 1997: 31-35.
- [4] Kühne F, Lages W F, GOMES da SILVA J M. Mobile robot trajectory tracking using model predictive control[C]//Proc of II IEEE Latin-American Robotics Symposium, Saint Louis: IEEE, 2005: 1-7.
- [5] Solea R, Filipescu A, Minzu V, et al. Sliding-mode trajectory-tracking control for a four-wheel-steering vehicle [C] // Proc of Control and Automation (ICCA), 2010 8th IEEE International Conference on. Xiamen: IEEE, 2010: 382-387.
- [6] Li Y, Wang Y. Trajectory tracking control of a redundantly actuated parallel robot using diagonal recurrent neural network[C]//Proc of Natural Computation, Fifth International Conference on. Tianjin: IEEE, 2009: 292-296.
- [7] Li Y, Wang Y, Chen Z. Research on trajectory tracking of a parallel robot based on neural network PID control[C]//Proc of Automation and Logistics, IEEE International Conference on. Qingdao: IEEE, 2008: 504-508.
- [8] Guo L, Ge P, Yang X, et al. Intelligent vehicle trajectory tracking based on neural networks sliding mode control[C]//Proc of Informative and Cybernetics for Computational Social Systems, 2014 International Conference on. Qingdao: IEEE, 2014: 57-62.
- [9] Kanjanawanishkul K. Motion control of a wheeled mobile robot using model predictive control: a survey [J]. KKU Res J, 2012, 17(5): 811-837.
- [10] 龚毅. 一种无人驾驶车辆路径跟踪控制方式研究 [D]. 南京: 南京理工大学计算机科学与工程学院, 2014.
- [11] Yi J, Zhang Y, Song D. Autonomous motorcycles for agile maneuvers, part I: Dynamic modeling [C]//Proc of Decision and Control, 28th Chinese Control Conference, Proceedings of the 48th IEEE Conference on. Shanghai: IEEE, 2009: 4613-4618.
- [12] Astrom K J, Klein R E, Lennartsson A. Bicycle dynamics and control; adapted bicycles for education and research[J]. Control Systems Magazine, IEEE, 2005, 25(4): 26-47.

.....

(上接第 314 页)

参 考 文 献

- [1] Canny J F. Constructing roadmaps of semi-algebraic sets I: completeness [J]. Artificial Intelligence, 1988; 37: 203-222.
- [2] Preparata F, Shamos M I. Computational geometry; an introduction [M]. Springer: Springer-Verlag, 1985.
- [3] Khatib O. Real-time obstacle avoidance for manipulators and mobile robots[J]. International Journal of Robotics Research, 1986, 5(1): 90-98.
- [4] Kavraki L E, Svestka P, Latombe J C, et al. Probabilistic roadmaps for fast path planning in high-dimensional configuration space[J]. Transactions on Robotics and Automation, 1996: 566-580.
- [5] Kuffner J J, LaValle S M. RRT-connect: an efficient approach to single-query path planning[C]//Proc of International Conference on Robotics and Automation. San Francisco: IEEE, 2000: 995-1001.
- [6] van den Berg J, Overmars M H. Planning the shortest safe path amidst unpredictably moving obstacles [C]//Proc of International Workshop on Algorithmic Foundations of Robotics. New York: IEEE, 2006: 885-897.
- [7] Leven P, Hutchinson S. Toward real-time path planning in changing environments [C] // Proc of the Fourth International Workshop on the Algorithmic Foundations of Robotics, 2000: 363-376.
- [8] Ding D, Liu H, Deng X, et al. A dynamic bridge builder to identify difficult regions for path planning in changing environments[C]//Proc of International on Robots and Systems. San Diego: IEEE, 2007: 2925-2931.
- [9] Ferguson D, Stentz A. Anytime dynamic planning in high-dimensional search spaces[C]//Proc of International Conference on Robotics and Automation. Kobe: IEEE, 2007: 1310-1315.
- [10] Zucker M, Kuffner J, Branicky M. Multipartite RRTs for rapid replanning in dynamic environments [C]// Proceedings of International Conference on Robotics and Automation. Roma: IEEE, 2007: 1603-1609.