

采用碰撞测试和回归机制的非完整约束机器人 快速扩展随机树运动规划

张波涛¹, 李加东², 刘士荣^{1†}

(1. 杭州电子科技大学 自动化学院, 浙江 杭州 310018; 2. 华东理工大学 自动化研究所, 上海 200237)

摘要: 本文提出一种改进的快速扩展随机树(rapidly-exploring random trees, RRT)运动规划方法, 用于非完整微分约束下的机器人运动规划. 针对类似目标偏好与双向RRT(bi-directional RRT, bi-RRT)等目标区域导向的RRT运动规划所存在的局部极小问题, 结合回归检测与碰撞检测机制, 设计了一种碰撞检测与回归机制(collision-test and regression mechanism, CR)机制. 该方法使得机器人在规划过程中能获取到全局障碍物信息, 从而避免对已扩展节点的重复搜索, 以及重复对边缘节点的回归测试和避障检测. 该机制使得机器人可加快跳出局部极小区域, 提高运动规划实时性. 将改进的RRT运动算法在容易产生局部极小值的环境中仿真测试, 结果表明该算法在不显著影响其他性能的前提下, 可以明显提高规划的实时性.

关键词: 机器人学; 运动规划; 非完整约束; RRT算法

中图分类号: TP242 **文献标识码:** A

Rapidly-exploring random trees motion planning for non-holonomic robot with collision-test and regression mechanism

ZHANG Bo-tao¹, LI Jia-dong², LIU Shi-rong^{1†}

(1. Institute of Automation, Hangzhou Dianzi University, Hangzhou Zhejiang 310018, China;
2. Institute of Automation, East China University of Science and Technology, Shanghai 200237, China)

Abstract: An improved rapidly-exploring random trees (RRT) algorithm is proposed to deal with the motion planning for non-holonomic mobile robots. The RRT algorithms using a bias towards the goal while choosing a random configuration, that will leads to the problem of local minima. Therefore, a novel method called collision-test and regression mechanism (CR) mechanism is presented, in which the collision detection mechanism and the regression testing mechanism are combined to enable the robot to escape from the local minima. The CR mechanism takes the global constraints into consideration, avoids exploring the directions which have been explored repeatedly. The repeatedly regression testing and detection for obstacle avoidance to the edge nodes are prevented in the CR. The ultimate goal of the algorithm is to improve the real-time performance of the planner, especially in the environment with highly-constraints. Simulation results of several improved RRT algorithms in the environment which is apt to generate local minima problems, verifies the proposed algorithm can improve the real-time performance significantly without obviously negative influences.

Key words: robotics; motion planning; non-holonomic constraint; RRT algorithm

1 引言(Introduction)

移动机器人的自主运动规划旨在为机器人在工作空间中规划出一组最优或次优的可行运动序列, 从而将复杂的任务分解为基本行为^[1]. 运动规划使机器人可以得到一条符合约束条件的运动序列, 约束条件与很多因素相关, 例如: 机器人的物理结构、运动规律、工作环境等. 具体情况而言, 速度限制、平衡问题、关

节受限以及躲避障碍物等问题均可构成对移动机器人的约束.

在早期研究中, 运动规划算法大多采用基于几何位形的构造方式, 这种方法的思路如下: 首先将特征位形空间映射为某种运动规划方法所需的离散图形, 通常工作位形空间与障碍物可以表达为以下3种形态: 单元分解图、道路图、虚拟人工势场. 在这些不同的

收稿日期: 2015-11-27; 录用日期: 2016-04-28.

† 通信作者. E-mail: liushirong@hdu.edu.cn; Tel.: +86 571-8687535.

本文责任编辑: 王伟.

国家自然科学基金项目(61503108, 61175093), 浙江省自然科学基金项目(LQ14F030012)资助.

Supported by National Natural Science Foundation of China (61503108, 61175093) and Natural Science Foundation of Zhejiang Province (LQ14F030012).

地图模型下, 可以采用各种不同的搜索算法规划最优或次优路径. 随着运动规划方法研究的不断深入, 一些具有强大搜索能力的智能运动规划算法, 以及其他随机路径搜索算法得到了移动机器人领域研究者的关注, 比如: 蚁群运动规划算法^[2]、RRT运动规划算法^[3]、人工势场法^[4]等.

移动机器人的约束可划分为非完整约束和完整约束, 若约束方程不能被积分为几何约束, 则其运动约束为非完整约束^[3]. 早期基于几何地图模型的传统运动规划方法, 以及部分智能规划方法对完整约束下的机器人规划有较大进展, 但是非完整约束的存在使得机器人运动规划变的更为复杂. 非完整约束下机器人不仅受到不可行区域导致的全局约束, 还受到自身运动机构的微分约束. 不同于全局约束, 这种自身结构导致的局部微分约束会对机器人的加速度与运动速度产生影响. 在非完整微分约束的运动规划中, 早期多数的运动规划方法与部分智能规划方法会面临维度灾难的限制, 环境规模的增大会导致运动规划方法的实时性非常糟糕. 其次, 此类运动规划方法所产生的运动轨迹并不符合移动机器人的运动学特点, 这会增加跟踪算法的复杂性, 甚至使得移动机器人难以将其应用于轨迹跟踪. 鉴于这些问题, 近年来, 机器人领域的研究者越来越关心基于非完整约束的运动规划方法^[4].

快速扩展随机树(rapidly-exploring random trees, RRT)算法是一种随机采样的搜索算法, 它通过随机扩展树向大面积未探索空间生长, 最终到达目标状态^[5-6]. RRT搜索算法非常适用于解决高维问题, 例如: 高自由度系统的运动规划问题, 这种优势使得RRT算法在近年来得到了国内外研究者的关注. 在机器人运动规划领域, RRT算法因其适用于非完整约束的特点, 引起了研究者的极大兴趣, 例如: 文献[7]针对自主导航车(automated guided vehicle, AGV)导航问题提出了一种定节点RRT算法, 该方法采用了增量逼近目标位形的方法, 用RRT在增量逼近过程中引导规划的方向, 并将其用于AGV的导航应用; 而文献[8]则针对动力学的微分约束提出了一种动态环境下的运动规划方法, 这种方法以栅格环境模型为基础, 对机器人工作空间的障碍物分布进行统计评估, 从而为RRT规划提供决策信息, 进而提高规划算法的搜索性能; 基于变换的快速扩展随机树(transition based rapidly-exploring random trees, T-RRT)方法则是将模拟退火算法与RRT运动规划算法相结合, 以此处理机器人在非平坦地势下的路径规划, 构建了基于机械功耗的最优化指标评价函数, 以此评估起伏地形中功耗, 此函数可在非平坦地形中工作时, 优化机器人所需要的机械能耗^[9]; 切从快速扩展随机树(tangent

bundle rapidly-exploring random trees, TB-RRT)则是一种基于切从的RRT运动规划算法, 该方法适用于流形曲面下的非完整约束运动规划, 它利用切从构建随机树的生长机制, 以此提高RRT算法的搜索能力^[10]; 文献[11]则是针对不规则障碍物环境, 提出了一种基于连续曲率的规划方法, 将环境约束与非完整约束进行结合, 引入了目标采样函数与合理性指标函数, 将其用于智能车的运动规划; RRT不仅适用于已知环境下的运动规划, 也可以用未知环境下的移动机器人运动规划^[12]. RRT算法的应用不局限于机器人领域, 在其他领域也得到了广泛应用, 例如: 文献[13]中的蛋白质分析中的路径搜索, 该研究对RRT算法进行了改进, 并将RRT用于从蛋白质运动位剥离配合基的路径分析, 将蛋白质折叠映射为高维空间下的路径规划.

基于目标偏好的RRT规划方法^[14](goal-biasing-RRT)与双向RRT规划方法^[15](bi-RRT)均为典型的以目标区域为生长引导的快速RRT规划方法. 这两种改进方式可以明显提高RRT的搜索性能, 同时, 也存在明显的局部极小缺陷. 本研究将碰撞检测法与回归测试法^[16-17]相配合的碰撞测试和回归(collision-test and regression mechanism, CR)机制用于RRT规划, 进而构建了一种改进的非完整约束下的RRT运动规划方法, 这种改进的RRT算法为采用碰撞检测与回归机制的快速扩展随机树, 简称为RRT-CR(RRT with collision-test and regression mechanism). 与传统RRT规划算法不同, CR机制使得改进的算法可在规划时综合考虑到更大范围的障碍分布情况, 进而利用规划过程中获得的障碍物信息, 改进了传统RRT算法仅以欧氏距离评估筛选节点, 进而决定扩展方向的方式. 基于欧氏距离的方式缺乏对整体局势的判断力, 往往使得机器人过早逃避障碍, 进而可能延长路径长度, 甚至难以发现路径. RRT-CR运动规划算法记录了随机树在工作空间中扩展的状态信息, 以此避免对已搜索节点或方向的重复搜索. 在回归检测阶段, 也防止了针对边缘节点的反复测试和重复检测, 从而使得规划算法以更高的速度跳出局部极小区域. RRT-CR算法可在不影响规划算法其他表现的情况下, 显著提升运动规划算法的实时性.

2 非完整约束与RRT算法(Non-holonomic constraint and RRT algorithm)

2.1 非完整约束(Non-holonomic constraint)

非完整约束^[18]对于一般的受限系统

$$\begin{cases} \dot{\vec{q}} = g(t, \vec{q}, r), \\ g(t, \dot{\vec{q}}, \vec{q}) = 0, \end{cases} \quad (1)$$

式中: \vec{q} 为状态变量, $\vec{q} \in \mathbb{R}^n$; r 是输入变量, $\vec{r} \in \mathbb{R}^n$;

t 是时间变量, $g(t, \vec{q}, r)$ 是系统的运动约束, 若不存在 $f(t, \vec{q})$ 满足下面的条件:

$$\frac{df(t, \vec{q})}{dt} = g(t, \dot{\vec{q}}, \vec{q}), \quad (2)$$

则该系统称为非完整系统, 而 $g(t, \dot{\vec{q}}, \vec{q})$ 被称为非完整约束.

图1是一种典型的非完整轮式机器人系统的运动模型, 在工作空间中该机器人有3个自由度, 其位姿向量如下 $\vec{q} = (\theta, x, y)$; x 为机器人在全局坐标系的横坐标, y 为纵坐标, θ 是机器人坐标系与全局坐标系的夹角. 该轮式机器人的前后轮轴距为 L . 对本系统而言, 输入变量包括: 转向角 φ , 以及机器人的线速度 v . 当机器人转向角为 φ 时, 机器人的圆弧运动半径为 ρ .

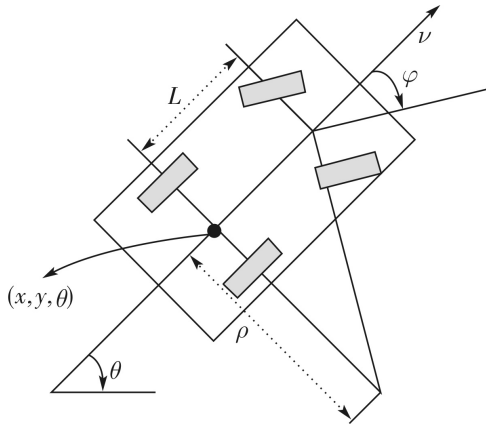


图1 典型轮式移动机器人模型

Fig. 1 Model of classical wheeled mobile robot

当 $\Delta t \rightarrow 0$ 时, 该移动机器人的微分约束可用式(3)表示:

$$\frac{dy}{dx} = \tan \theta. \quad (3)$$

因为 $\frac{\dot{y}}{\dot{x}} = \frac{dy}{dx}$, 故机器人轮子的微分约束可以转换为

$$\dot{y} \cos \theta - \dot{x} \sin \theta = 0. \quad (4)$$

由图1可知

$$L = \rho \cdot \tan \varphi, \quad (5)$$

$$d\theta = (\tan \varphi / L) \cdot di_v. \quad (6)$$

在上式中, i_v 为移动机器人速度的积分, 将式(5)与式(6)的两边分别对时间求导, 整理后, 可得下式:

$$\dot{\theta} = (\tan \varphi \cdot v) / L, \quad (7)$$

令 $r = (r_v, r_\varphi)^T$, 根据公式(7)以及 \dot{x} 与车体速度的关系, 可得状态转移方程如下:

$$\begin{cases} \dot{x} = \cos \theta \cdot r_v, \\ \dot{y} = \sin \theta \cdot r_v, \\ \dot{\theta} = (\tan(r_\varphi) \cdot r_v) / L. \end{cases} \quad (8)$$

移动机器人系统为时间连续的非完整约束系统, 其中 $\vec{q} = (x, y, \theta)$, 式(8)可变化如下形式:

$$\vec{q}_{\text{new}} \approx \int_0^{\delta t} g(\vec{q}, r) dt + \vec{q}. \quad (9)$$

2.2 RRT算法(RRT algorithm)

图2为RRT扩展过程示意图, RRT基本原理为: 在初始化的过程中, 仅将初始状态点加入到随机树中, 而在接下来的迭代过程中不断扩展随机树, 其伪代码如下, 其中Select(\cdot)函数用来选取最佳的输入组合:

RRT_Pseudo(\cdot)

```

1   Treeinitial( $q_{\text{initial}}$ );
2   while! flag
3        $q_{\text{rand}} \leftarrow \text{Point\_sample}(\cdot)$ ;
4        $q_{\text{nearest}} \leftarrow \text{Node}_{\text{nearest}}(q_{\text{rand}}, T)$ ;
5        $q_{\text{new}}, u_{\text{best}} \leftarrow \text{Select}(q_{\text{rand}}, q_{\text{nearest}})$ ;
6       if (free of collision)
7           tree  $\leftarrow \{q_{\text{new}}\} \cup \text{tree}$ ;
8       if  $\|q_{\text{goal}} - q_{\text{new}}\| \leq d$ 
9           flag = true;
10      else flag = false.
```

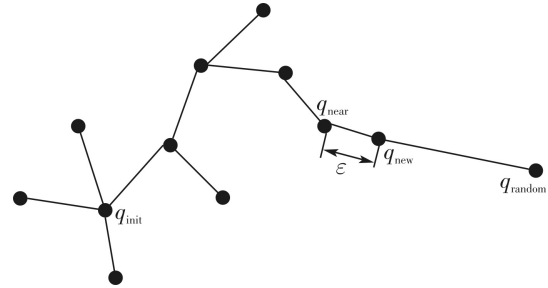


图2 RRT的扩展过程

Fig. 2 Expansion process of RRT

2.3 目标区域导向的RRT(Goal area oriented RRT)

RRT运动规划方法的随机搜索机制使该方法在高维问题中具有较好的表现, 同时, RRT的随机性使得算法在位姿空间中的采样服从均匀分布, 从而生成下一步的节点. 这种随机搜索的机制降低了随机树在工作空间中的搜索速度, 也降低了所获得路径的品质. 针对此问题, 一些研究者对RRT规划方法做了一些改进, 旨在提高RRT规划方法的搜索实时性, 同时, 保持其随机搜索能力, 例如: 基于目标偏好的RRT方法(goal biasing RRT)^[14]与双向RRT(bi-directional RRT, bi-RRT)^[15].

基于目标偏好的RRT算法(goal biasing RRT)在主搜索树扩展中对目标点 q_{rand} 采用随机取样, 而以极小的概率从目标区域中生成位姿状态, 以此替换随机目标位姿, 从而引导搜索树向目标区域靠近. 基于目标

偏好的RRT算法旨在用目标区域内的随机状态引导搜索树的生长过程, 其伪代码如下:

```

RRTGoal-Biasing(·)
1   Point_sample(·)
2    $T = \text{rand\_number} \in (0, 1)$ 
3   if  $T > T_{\text{threshold}}$ 
4      $q_{\text{rand}} = \text{Rand\_state}(\cdot)$ ;
5   else
6      $q_{\text{goal}} = q_{\text{rand}}$ ;

```

由上可见: 选取RRT目标区域的随机目标位姿时, 概率 T 服从均匀分布, 设置概率阈值 $T_{\text{threshold}}$, 该阈值服从以下规律: $T_{\text{threshold}} \in [0.05, 0.1]$. 若 T 大于阈值 $T_{\text{threshold}}$, 则随机目标位姿从目标区域的空间随机获取; 否则, $T_{\text{threshold}}$, 则以位姿 q_{goal} 成为RRT随机树下一步生长的目标.

bi-RRT运动规划算法(bi-RRT motion planning algorithm)属于另一种基于目标区域导向的RRT算法, 该算法采用2个随机树从初试位姿与目标位姿相向生长. 其中1棵随机树向目标位姿生长, 第2棵随机树以第1棵随机树新生成的节点作为目标位姿生长扩展, 在第2次迭代时, 需要变更两棵随机树的生长顺序, bi-RRT规划算法的伪代码如下:

```

bi-RRT(·)
1    $\text{Tree}_{d1}, \text{Tree}_{d2} \leftarrow \text{Tree}_{\text{initial}}(q_{\text{goal}}, q_{\text{initial}})$ ;
2   while! flag
3     flag = false;
4      $q_{\text{rand}} \leftarrow \text{Point\_sample}(\cdot)$ ;
5      $q_1 \leftarrow \text{Tree\_grow}(q_{\text{rand}}, \text{Tree}_{d1})$ ;
6     if  $q_1$  turn
7        $q_2 \leftarrow \text{Tree\_grow}(q_1, \text{Tree}_{d2})$ ;
8     if  $q_2$  turn
9        $\text{Tree}_{d1} \leftarrow \text{Tree}_{d1} + q_1$ ;
10       $\text{Tree}_{d2} \leftarrow \text{Tree}_{d2} + q_2$ ;
11     if  $\|q_1 - q_2\| < \lambda$ 
12       flag = true;
13    $\text{Tree}_{d1} \Leftrightarrow \text{Tree}_{d2}$ .

```

$\text{Tree_grow}(q, T)$ 表示以 q 为目标点, 拓展随机树 T . $\text{Tree}_{d1} \Leftrightarrow \text{Tree}_{d2}$ 表示将两棵随机树进行交换, λ 表示一个极小的正常数.

RRT运动规划算法无记忆功能, 算法会重复对已经探索区域重复探索. 基于目标偏好的RRT与bi-RRT均以目标点所在区域为引导, 使随机树向目标区域生长. 这种规划方式非常像APF规划, 均以梯度下降的方式导向目标, 且都存在局部极小. 纯粹的加速生长方法均可能在一些局部区域产生问题. 在复杂工作环境中, 这种局部区域存在的可能性高, 引起局部极小的可能性也较大. 如果进入局部极小区域, RRT规划算法会在部分局部复杂区域中重复扩展, 耗时过长.

3 RRT-CR 运动规划 (RRT-CR motion planning)

3.1 回归测试的原理 (The principle of regression testing)

据上面所述, 目标区域引导的快速RRT规划均存在局部极小现象, 文献[17]提出了一种回归测试方法, 以此降低算法陷入局部极小区域的概率, 并将其与RRT结合, 从而构建了一种涌现式快速扩展随机树 (rapidly-exploring random trees-blossom, RRT-Blossom)算法. 回归检测方法的主要机理如下: 在可能存在局部极小的区域, 随机树可以向四周探索, 并可以倒退反向生长, 以此向未拓展过的环境进行生长. 该方法构建了一种过滤节点的机制: 先判断在最近的邻边缘节点位姿态中, 算法是否生长到曾经被搜索过的区域. 若已经探测过, 则该节点被设置为回归点, 并淘汰出局, 然后从向未探测区域生长过的节点集合中选择最近的节点. 判别区域 E 是否曾经被搜索过的方法如下: 当新生成的节点 q_{new} 与随机树上最近的节点之间的欧式距离小于 q_{new} 与母节点 q_{parent} 之间的距离时, 则区域曾经被搜索过, 具体如式(10)所示:

$$\begin{cases} \exists q \in \text{tree}, \\ \|q_{\text{parent}}, q_{\text{new}}\| > \|q, q_{\text{new}}\|. \end{cases} \quad (10)$$

RRT-Blossom运动规划算法仍然存在一些问题, 据前面所述, 经典RRT规划方法搜索环境时, 节点和搜索方向会被不断反复的尝试搜索, 这种问题仍然存在于RRT-Blossom中, 具体表现为在每次迭代搜索中, 算法会对潜在的边缘化节点反复进行回归检测与故障测试, 会严重降低规划算法的执行效率. 同时, 该运动规划方法在一些狭窄的可行通道中无法得到有效路径, 其根本原因是: 本来可以通过狭窄通道的节点有一定概率会因距离随机树上的其他无效节点太近, 而被算法淘汰.

无论RRT-Blossom还是经典RRT算法, 均利用欧氏距离筛选最佳输入与最近邻节点, 该办法并没有综合考虑到全局的环境信息.

3.2 碰撞检测机制的原理 (The principle of detection for obstacle avoidance)

针对文献[17]中所提算法中回归测试所存在的一些问题, 本文设计了一种碰撞检测方法, 结合原有的回归测试, 构建了一种新的CR检测机制. 该机制使得RRT算法可以用更少的时间脱离局部极小区域, 以此提高规划算法的实时性.

规划算法的碰撞检测与采集的示意图如图3所示, 设工作空间中状态点的输入集合的维度为 n , 随机树的初试点 q_{init} 位于顶端, 其后是各层节点的状态. 假设 σ 为各节点与障碍的碰撞概率, 初始状态时, 新节点

的初试碰撞概率为0, 该节点各方向被初始化为未探索状态. 图3中: 若 q_{nearest} 在该节点的子方向与障碍物发生碰撞, 则该方向会被重置为已探索状态, 与此同时 q_{nearest} 所对应的 σ 值会提高 $1/m$, 而其最近的邻节点位姿的母节点对应的碰撞概率相应提高 $1/m^2$, 而此邻节点前面的 k 阶母节点所对应的碰撞概率提高 $1/m^{k+1}$, 直到将各节点碰撞概率计算到初始节点.

各节点对应的 σ 值可以描述全局碰撞约束的状况, 其中, $\sigma = 1$ 则意味着该节点各输入方向均会导致碰撞, 而该节点会因此失去被再次生长的价值, 而 σ 值越小的节点则意味着较高的拓展价值, 会有更大的概率被优选为生长节点. 生成最近的邻节点时, 需要综合计算碰撞几率, 若某节点的碰撞概率较大, 则表示该节点各方向有很大可能会遇到障碍物, 尽量选择那些碰撞概率最低的节点为生长方向, 这有利于引导随机树在搜索过程中远离障碍物. 一旦 q_{nearest} 被确定, 最佳输入从该节点尚未搜索的方向中挑选, 而 σ 值最小的节点选择的对象. 此后, 利用碰撞检测和回归测试评估新产生的状态点是否符合要求.

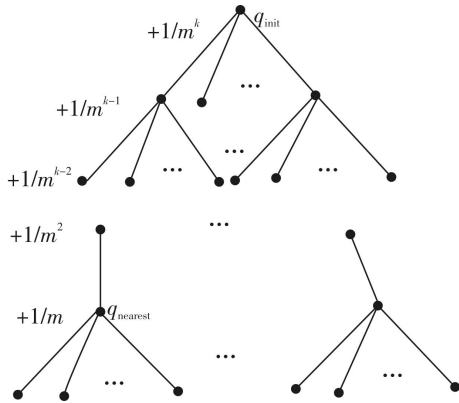


图3 碰撞信息采集

Fig. 3 Collision-information collecting

碰撞信息采集函数 $\text{Collision_inf}(\cdot)$ 与回归检测函数 $\text{Regress_test}(\cdot)$ 伪代码如下所示, 其中函数 $\text{Regress_test}(\cdot)$ 以式(10)为基础.

$\text{Regress_test}(\text{Tree}, q_{\text{new}}, q_{\text{parent}})$

```

1   for  $q_{\text{tree}} \in \text{tree}$ 
2     if  $\|q_{\text{new}}, q_{\text{tree}}\| < \|q_{\text{new}}, q_{\text{parent}}\|$ 
3       return  $T$ ;
4     else
5       return  $F$ ;

```

$\text{Collision_inf}(\text{Tree}, q_{\text{nearest}})$

```

1    $\lambda \leftarrow 1$ ;
2    $q \leftarrow q_{\text{nearest}}$ ;
3   while  $q$ 
4      $\lambda \leftarrow \lambda/m$ ;
5      $\sigma \leftarrow \lambda + \sigma$ ;
6      $q \leftarrow \text{parent node of } q$ .

```

碰撞检测对RRT规划的主体结构并无影响, 发生变化的是最佳输入的选择以及最近邻节点的选择, 二者的伪代码为 $\text{Node}_{\text{nearest}}(q_{\text{rand}}, \text{tree})$, $\text{Select}(q_{\text{rand}}, q_{\text{nearest}}, \text{tree})$.

Nearest node and input selecting

```

1    $\text{Node}_{\text{nearest}}(q_{\text{rand}}, \text{tree})$ 
2      $d_{\text{min}} \leftarrow \infty$ ;  $d \leftarrow \infty$ ;
3     for  $\forall q \in Q_{\text{unsearched}}$ 
4        $d \leftarrow \|q_{\text{rand}}, q\|$ ;
5       if  $d_{\text{min}} > d$ 
6          $d_{\text{min}} \leftarrow d$ ;
7         Nearest  $\leftarrow q$ ;
8    $\text{Select}(q_{\text{rand}}, q_{\text{nearest}}, \text{tree})$ 
9      $u_{\text{best}} \leftarrow \emptyset$ ,  $d_{\text{min}} \leftarrow \|q_2 - q_1\|$ ;
10    for  $u \in U$ 
11      if  $u \in U_{\text{unsearched}}$ 
12         $q_{\text{new}} \leftarrow \text{Test}_{\text{integrate}}(u, q_{\text{nearest}})$ ;
13        if  $\text{Regress\_test}(q_{\text{new}}, q_{\text{parent}}, \text{tree})$ 
14           $U_{\text{searched}} \leftarrow u$ ;
15          test  $u_{k+1}$ 
16        else
17          if  $\text{Disable}(q_{\text{new}}, q_{\text{parent}}, u)$ 
18            test  $u_{k+1}$ ;
19             $\text{Collision\_inf}(q_{\text{nearest}}, \text{tree})$ ;
20             $d \leftarrow \|q_{\text{rand}} - q_{\text{new}}\|$ ;
21            if  $d_{\text{min}} > d$ 
22               $u_{\text{best}} \leftarrow d, d_{\text{min}}, u$ 
23    Return  $u_{\text{best}}, q_{\text{new}}$ .

```

在生成最近的邻节点时, 需要从尚未搜索过的节点中最近者, 其中, 该节点集合含有碰撞概率小于 $T_{\text{threshold}}$, 且该节点仍然有方向尚未被搜索; 最佳输入的选择与经典RRT规划的对应功能模块也存在一定不同, 碰撞信息的存在便于规划算法挑选更优的输入. 针对任一输入条件, 若这个输入的方向所生成的新节点导致碰撞或者回归, 则该输入条件将被记录为已搜索, 以此避免冗余计算. 此后, 还需要对最近的邻节点及其前面的各阶母节点更新碰撞概率值. 碰撞检测机制的时间复杂度为 $O(n)$, 其空间复杂度为 $O(n)$, RRT-CR运动规划算法的空间复杂度为 $O(n)$, 时间复杂度为 $O(n \log(n))$.

4 仿真实验与分析(Simulation experiments and analysis)

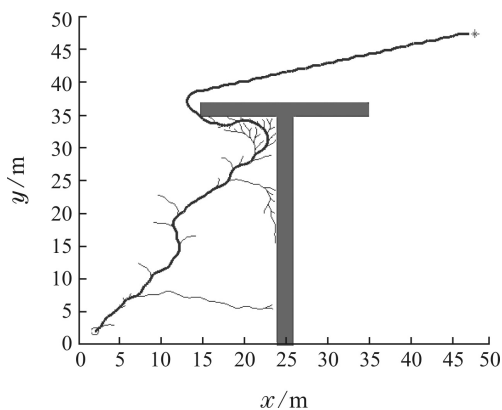
为了测试碰撞检测与回归测试相结合的CR机制在运动规划中的表现, 选择图1所示的非完整约束轮式移动机器人为载体, 将几种搜索机制分别用于bi-RRT运动规划, 以及基于目标偏好的RRT运动规划中, 并选择若干易于导致局部极小的环境模型进行仿真测试. 对于bi-RRT算法和基于目标偏好的RRT所存

的易于陷入局部极小的问题, 本研究采用障碍物比较密集的环境模型、T型障碍检测基于CR机制的改进算法的性能表现; 同时, 针对RRT-Blossom规划中回归测试机制在狭窄通道存在的瓶颈, 本研究构建了两种不同宽度的狭窄通道, 并在其中进行仿真, 以此检验改进算法在不同狭窄通道中的表现。

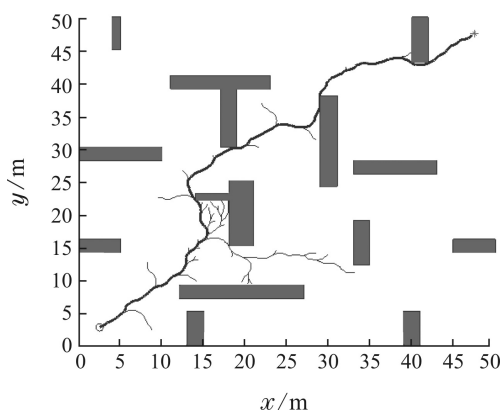
仿真配置如下: 机器人的线速度 v 为3 m/s, $\varphi \in (-\pi/6, \pi/6)$, 本体的参数 L 取0.5 m, 采样间隔 $\Delta t = 0.3$ s. 控制输入集与最近邻节点参照式(8), 在 Δt 上, 以式(9)更新的机器人位姿 q_{new} . 仿真计算机配置如下所示: 处理器 Intel Core(TM)2(2.10 GHz), 内存 2.00 G, 仿真环境则采用MATLAB R2010b, 以下仿真环境中的横、纵坐标分别对应 \bar{q} 中的元素 x 与 y , 单位均为m.

4.1 仿真实验(Simulation experiments)

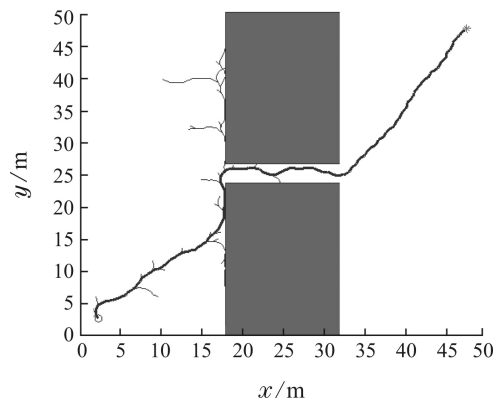
机器人分别采用回归测试机制和CR机制与目标偏好RRT运动规划相结合, 并在T型障碍环境模型(图4(a))、非规则障碍环境模型(图4(b))、狭窄通道A(图4(c))和狭窄通道B(图4(d))进行仿真测试, 以上测试所得结果如图4所示, 相关测试数据的统计如表1所示. 表1中数据是经独立的50次实验运行所得数据的均值, 其中: N_{node} 表示平均节点数目, L_{road} 为平均路径长度, T_{plan} 表示平均规划时间.



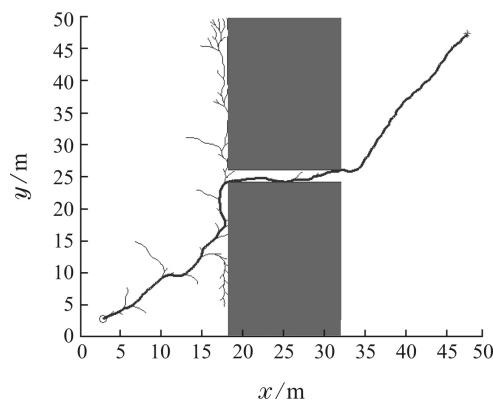
(a) T型障碍环境模型



(b) 非规则障碍环境模型



(c) 狭窄通道A



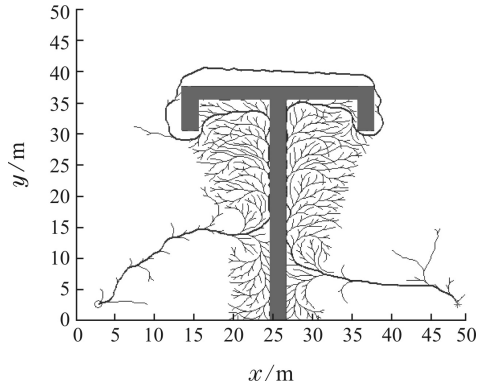
(d) 狭窄通道B

图 4 基于改进目标偏好的RRT规划
Fig. 4 Improved RRT planning based on goal biasing

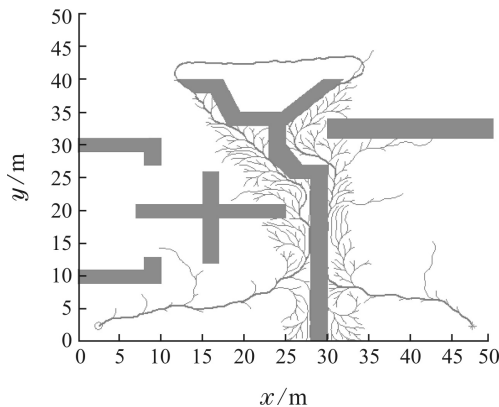
表 1 基于改进目标偏好的RRT规划
Table 1 Improved RRT planning based on goal biasing

| 环境 | 机制 | 成功率/% | N_{node} | L_{road} | T_{plan} |
|-------|------|-------|------------|------------|------------|
| T型障碍 | 回归测试 | 100 | 245 | 91.566 | 7.098 |
| | CR | 100 | 244 | 91.674 | 1.251 |
| 非规则障碍 | 回归测试 | 100 | 197 | 74.268 | 3.266 |
| | CR | 100 | 200 | 75.870 | 1.217 |
| 狭窄通道A | 回归测试 | 89 | 138 | 70.902 | 1.106 |
| | CR | 90 | 111 | 72.288 | 0.858 |
| 狭窄通道B | 回归测试 | 71 | 132 | 71.424 | 1.304 |
| | CR | 65 | 140 | 72.342 | 0.856 |

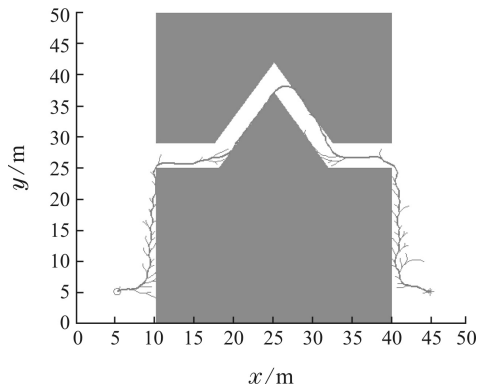
如图5所示, 移动机器人分别采用回归测试机制和CR机制与bi-RRT运动规划算法进行结合, 并在T型障碍环境模型(图5(a))、非规则障碍环境模型(图5(b))、狭窄通道C(图5(c))和狭窄通道D(图5(d))中分别进行仿真测试. 以上仿真测试的结果如图5所示, 相关测试数据的统计如表2所示, 表2中数据是经50次运行所得数据的均值.



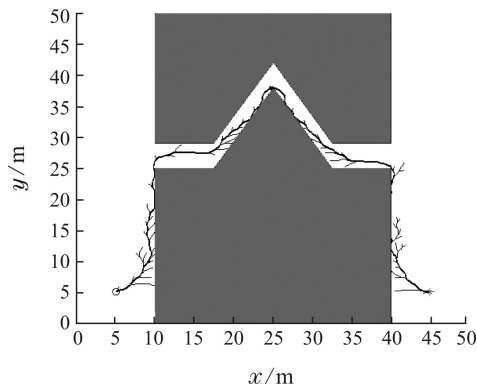
(a) T型障碍环境模型



(b) 非规则障碍环境模型



(c) 狭窄通道C



(d) 狭窄通道D

图5 改进的bi-RRT运动规划

Fig. 5 Improved bi-RRT motion planning

表2 改进bi-RRT规划的统计数据

Table 2 Statistical data of improved bi-RRT planning

| 环境 | 机制 | 成功率/% | N_{node} | L_{road} | T_{plan} |
|-------|------|-------|-------------------|-------------------|-------------------|
| T型障碍 | 回归测试 | 100 | 1239 | 161.966 | 28.367 |
| | CR | 100 | 1253 | 162.562 | 18.289 |
| 非规则障碍 | 回归测试 | 100 | 818 | 146.851 | 8.880 |
| | CR | 100 | 787 | 147.097 | 6.217 |
| 狭窄通道C | 回归测试 | 96 | 200 | 100.577 | 1.386 |
| | CR | 94 | 192 | 92.596 | 1.304 |
| 狭窄通道D | 回归测试 | 89 | 207 | 102.161 | 1.336 |
| | CR | 89 | 181 | 93.570 | 1.106 |

4.2 结果分析(Result analysis)

bi-RRT运动规划与基于目标偏好的RRT运动规划因为在随机生长过程中存在目标区域的导向作用, 因此, 必然其在某些情况下会产生局部极小的陷阱. 在以上测试中, 将碰撞检测与回归检测相结合的CR机制用于bi-RRT运动规划与基于目标偏好的RRT运动规划, 试图让规划算法可以更快的脱离局部极小, 具有更强的实时性. 将基于CR机制的RRT算法与单纯采用回归检测的RRT算法进行比较得到如下结论.

1) 由图4与表1可见, 无论是将CR机制, 还是将回归测试应用于基于目标偏好的RRT规划, 规划所获得的路径长度、节点数量、成功率都极为相近, 然而, 基于CR机制的规划算法在实时性方面极具优势. 基于CR机制的RRT与单纯基于回归检测的RRT运动规划相比, 前者所消耗的时间只有后者所需时间的17.62%, 37.26%, 77.6%与65.64%. 此外, 由表1可见, 存在类似T型障碍的这种深度极小环境的模型中, CR机制在实时性方面具有极大的优势.

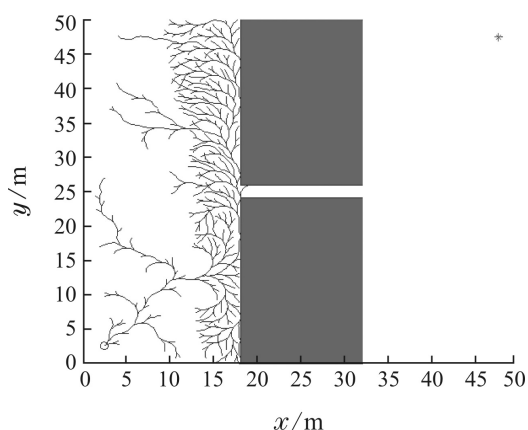
2) 由图5与表2可见, 无论将CR机制还是将回归测试应用于bi-RRT规划, 所得路径长度、节点数量、成功率仍然都非常相似. 与单纯基于回归检测的bi-RRT运动规划相比, 基于CR机制的bi-RRT规划耗时明显减少. 在4类不同环境中, CR机制的耗时只有回归检测机制的64.48%, 70.01%, 94.08%, 以及82.78%. 环境越复杂, 则CR机制的实时性优势越明显, 由此可见, CR机制更适合处理复杂环境下的非完整约束运动规划问题.

碰撞检测方法弥补了RRT运动规划仅以欧氏距离进行节点选择时, 随机树的生长过程缺乏对全局约束

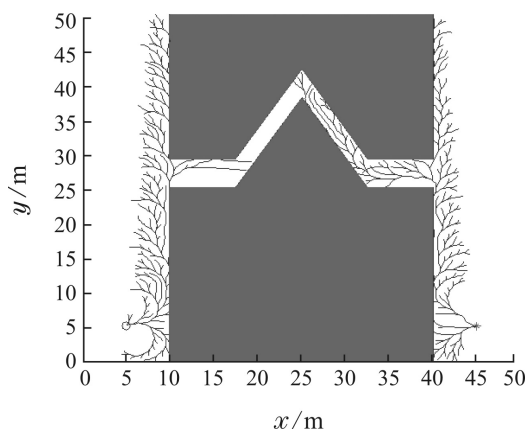
的考虑. 该机制可以避免对已探索方向和节点的反复探索, 在回归检测中, 该机制也可以防止对边缘节点的反复测试. 回归测试与碰撞检测的组合作用使得基于CR机制的随机树可以更快的跳出局部极小陷阱, 从而使RRT算法的实时性得到了显著改善.

图6(a)为基于CR的目标偏好RRT算法, 图6(b)为基于CR的bi-RRT算法. 由表1、表2以及图6可见, 与bi-RRT与目标偏好RRT一样, 基于CR机制的RRT运动规划依然无法完全解决狭窄通道的问题, 这几种算法在狭窄通道中的通过率均无法达到100%. CR机制的主要作用是提高实时性, 无法完全避免狭窄通道导致的规划失败.

规划失败的原因如下: 在狭窄通道生成有效节点的概率很低, 即使偶尔生成有效节点, 也可能会因其距离随机树上的其他不合理节点过近, 而被淘汰. 受内存所限制, 随机搜索算法的迭代次数是有限的, 无法在有限的时间内获得达到目标位置的有效节点, 则会导致搜索失败. 通过增加迭代次数, 可以得到更高的狭窄通道成功率.



(a) 基于CR的目标偏好RRT算法



(b) 基于CR的bi-RRT算法

图6 CR机制中的问题

Fig. 6 Problems in CR

5 结论(Conclusions)

运动规划是移动机器人的关键功能模块之一, 而非完整约束则是运动规划中最常见的约束. 针对非完整约束下的机器人运动规划, 本文将回归检测与碰撞检测相结合构建了CR机制, 并将其用于两种目标区域导向的RRT算法, 以此提高原RRT规划算法的实时性, 结果证明CR机制可以在不降低RRT算法其他性能的前提下, 明显改善RRT运动规划算法的实时性.

参考文献(References):

- [1] LAVALLE S M. *Planning Algorithm* [M]. Cambridge: Cambridge University Press, 2006.
- [2] ZHAO Juanping, GAO Xianwen, FU Xiuhui, et al. Improve ant colony algorithm of path planning for mobile robot [J]. *Control Theory & Applications*, 2011, 28(4): 448 – 452. (赵娟平, 高宪文, 符秀辉, 等. 移动机器人路径规划的改进蚁群优化算法 [J]. 控制理论与应用, 2011, 28(4): 448 – 452.)
- [3] ERION P. Region-guided and sampling-based tree search for motion planning with dynamics [J]. *IEEE Transactions on Robotics*, 2015, 31(3): 723 – 735.
- [4] ZHU Yi, ZHANG Tao, SONG Jingyan. Path planning for nonholonomic mobile robots using artificial potential field method [J]. *Control Theory & Applications*, 2010, 27(2): 152 – 158. (朱毅, 张涛, 宋靖雁. 非完整移动机器人的人工势场法路径规划 [J]. 控制理论与应用, 2010, 27(2): 152 – 158.)
- [5] LAVALLE S M. *Rapidly-exploring random trees: A new tool for path planning* [R]. Iowa: Computer Science Department, Iowa State University, 1998.
- [6] SPANOGLIANOPOULOS S, SIRLANTZIS K. Non-holonomic path planning of car-like robot using RRT * FN [C] // *Proceedings of IEEE International Conference on Ubiquitous Robots and Ambient Intelligence*. New York: IEEE, 2015, 10: 53 – 57.
- [7] GUITTON J, FARGES J L, CHATILA R. Cell-RRT: Decomposing the environment for better plan [C] // *Proceedings of International on IROS*. New York: IEEE, 2009, 10: 5776 – 5781.
- [8] SUCAN I A, KAVRAKI L E. A sampling-based tree planner for systems with complex dynamics [J]. *IEEE Transactions on Robotics*, 2012, 28(1): 116 – 131.
- [9] JAILLET L, CORTES J, SIMEON T. Sampling-based path planning on configuration-space costmap [J]. *IEEE Transactions on Robotics*, 2010, 26(4): 647 – 659.
- [10] KIM B, UM T T, SUH C, et al. Tangent bundle RRT: A randomized algorithm for constrained motion planning [J]. *Robotica*, 2016, 34(1): 202 – 225.
- [11] DU Mingbo, MEI Tao, CHEN Jiajia, et al. RRT-based motion planning algorithm for intelligent vehicle in complex environments [J]. *Robot*, 2015, 37(4): 443 – 450. (杜明博, 梅涛, 陈佳佳, 等. 复杂环境下基于RRT的智能车辆运动规划算法 [J]. 机器人, 2015, 37(4): 443 – 450.)
- [12] DUGAN U, PUTRA H C. Manipulator path planning in unknown environment using model based planners: conversion criteria and IPA sensor implementation [C] // *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*. New York: IEEE, 2009, 10: 273 – 278.

- [13] CORTES J, JAILLET L, SIMEON T. Molecular disassembly with RRT-like algorithms [C] // *Proceedings of IEEE International Conference on Robotics and Automation*. New York: IEEE, 2007, 4: 3301 – 3306.
- [14] URMSON C, SIMMONS R. Approaches for heuristically biasing RRT growth [C] // *Proceedings of IEEE Conference on Intelligent Robots and Systems*. New York: IEEE, 2003, 10: 1178 – 1183.
- [15] HSIEN L, CHENG SUN Y. 2D-span resampling of bi-RRT in dynamic path planning [J]. *International Journal of Automation and Smart Technology*, 2015, 5(1): 39 – 48.
- [16] LAVALLE S M, KUFFNER J J. Randomized kino-dynamic planning [J]. *International Journal of Robotics Research*, 2001, 20(5): 378 – 400.
- [17] KALISIAK M, PANNE M V. RRT-Blossom: RRT with a local flood-fill behavior [C] // *Proceedings of IEEE International Conference on Robotics and Automation*. New York: IEEE, 2006, 5: 1237 – 1242.
- [18] RICHARD M, ZEXIANG L, SASTRY S S. *A Mathematical Introduction to Robotic Manipulation* [M]. Florida: CRC Press, 1994.

作者简介:

张波涛 (1982-), 男, 讲师, 博士, 目前研究方向为机器人运动规划与控制、图像处理等, E-mail: billow@hdu.edu.cn;

李加东 (1989-), 男, 硕士, 目前研究方向为机器人运动规划与导航, E-mail: lijiaodong106305@126.com.cn;

刘士荣 (1952-), 男, 教授, 博士生导师, 目前研究方向为复杂系统建模、智能系统与智能机器人等, E-mail: liushirong@hdu.edu.cn.