

文章编号: 2095-2163(2022)01-0016-05

中图分类号: TP242.6

文献标志码: A

基于节点过滤及运动学约束改进的 RRT 算法

孟祥永, 游彩霞, 严运兵

(武汉科技大学汽车与交通工程学院, 武汉 430070)

摘要: 由于快速扩展随机树(RRT)算法存在规划路径曲折、收敛较慢,且无法被智能车辆直接跟踪等问题。为克服此缺陷,本文在基本 RRT 算法基础上,加入目标偏向性策略和密集节点过滤,以此提高规划速度;在选择父节点时考虑车辆运动学约束并根据车辆位置动态确定扩展步长,最后对所得路径进行修剪,并使用三次 B 样条曲线进行平滑,生成一条平滑可被追踪的曲线。仿真实验表明,改进的 RRT 算法生成路径更加合理、平滑,且符合车辆运动学特性。

关键词: 智能车; 路径规划; 快速扩展随机树; 运动学约束

Improved intelligent vehicle path planning RRT algorithm based on node filtering and kinematic constraints

MENG Xiangyong, YOU Caixia, YAN Yunbing

(School of Automotive and Traffic Engineering, Wuhan University of Science and Technology, Wuhan 430070, China)

[Abstract] Rapid Extended Random Tree (RRT) algorithm planning path twists and turns, convergence is slow, and can not be directly tracked by intelligent vehicles. To overcome this defect, an improved RRT algorithm is proposed in this paper. Based on the basic RRT algorithm, target bias strategy and dense node filtering are added to improve the planning speed. When selecting the parent node, the vehicle kinematics constraint is considered and the extended step size is determined dynamically according to the vehicle position. Finally, the path obtained is trimmed and smoothing by cubic B-spline curve to generate a smooth and tractable curve. The simulation results show that the path generated by the improved RRT algorithm is more reasonable and smooth, and conforms to the vehicle kinematics characteristics.

[Key words] intelligent vehicle; path planning; Rapidly-Exploring Random Tree; vehicle kinematic constraint

0 引言

为了提高出行效率和保障行车安全,辅助驾驶与自动驾驶技术逐渐成为研究热点。路径规划是智能车行驶的必要部分,其主要作用是已知车辆当前位置、目标点位置和外部环境情况下,规划出一条符合车辆运动学要求且安全无碰撞的路径^[1-2]。相关学者提出并改进了各种规划算法,主要包括图搜索法^[3]、人工势场法^[4]和采样法^[5]等。

图搜索法生成的轨迹难以满足车辆的运动学约束,且需要对场景离散化;人工势场法将环境抽象为力场函数,虽简单高效,但容易陷入局部最优。文献[6]提出的快速搜索随机树,是一种基于采样的路径规划算法。通过对工作空间进行随机采样,无需对状态空间进行建模,同时可以考虑智能车运动过程中的各种约束,因此得到广泛应用。基本的 RRT

规划算法存在的不足:

- (1) 采样点随机,收敛结果非最优解;
- (2) 生成的路径曲折,车辆无法直接跟踪;
- (3) 搜索过程无导向性^[7-8]。

针对基本 RRT 算法的缺点,国内外学者进行了大量的改进研究。文献[9]提出的 RRT* 算法,相较基本 RRT 的主要改进是在随机树扩展时,对父节点的重新考量,寻到终点后不会立即停止,随着采样点增多,所得路径也会逐渐优化,但收敛速率较慢。文献[10]中提出的目标偏向搜索和动态步长虽然提高了求解速率,但并未克服路径曲折及采样点随机的的问题。文献[11]提出的 TG-RRT 算法,将目标点、起始点和随机生成的点连接成三角形,用该三角形的形心或内心替代随机点,以此缩小搜索范围,减少搜索时间;文献[12]使用 B 样条曲线对路径点进行拟合,可以保证拟合路径曲率的连续性,且满足车

基金项目: 国家自然科学基金(51975428)。

作者简介: 孟祥永(1996-),男,硕士研究生,主要研究方向:智能驾驶运动规划;游彩霞(1978-),女,博士,副教授,主要研究方向:汽车动力学、汽车 NVH;严运兵(1968-),男,博士,教授,主要研究方向:新能源汽车及其控制、汽车系统动力学。

通讯作者: 严运兵 Email: yangyunbing@wust.edu.cn

收稿日期: 2021-10-08

辆运动学要求。

上述算法虽然在一定程度上改进了 RRT 算法,但车辆运行环境极其复杂,不仅要提高收敛速度外,还要考虑实用性。本文针对采样点随机、路径曲折等问题,提出一种改进 RRT 算法;其包含目标偏向策略、密集节点过滤、基于车辆运动学约束父节点选择,随机步长动态扩展,并对规划得到的路径进行冗余节点修剪和基于三次 B 样条曲线进行平滑,最后通过仿真实验证明了算法的有效性。

1 基本 RRT 算法

基本 RRT 算法是以初始位置 q_{rand} 为根节点,通过随机生成采样点的方式进行迭代扩展,判断新的采样点是否到达目标范围内,直到该条件满足,在随机树节点集中,从终点向起点寻找仅有的一条路径。如图 1 所示。

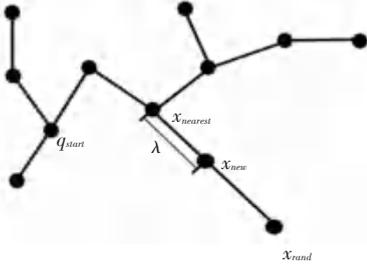


图 1 随机树扩展示意图

Fig. 1 Random tree expansion diagram

基本 RRT 算法实现步骤如下:

步骤 1 搜索空间及算法相关参数初始化。

步骤 2 判断是否超过迭代次数,超过寻路失败。

步骤 3 产生一随机点 x_{rand} , 遍历节点集 T 寻找其最近的节点 $x_{nearest}$ 作为其父节点。

步骤 4 以父节点 $x_{nearest}$ 为根向随机点 x_{rand} 方向扩展一定的步长 λ , 得到新节点 x_{new} 。

步骤 5 若新节点 x_{new} 无碰撞, 则将产生的新节点加入节点集中; 判断该节点是否在目标范围内, 成立则结束搜索; 反之返回步骤 2。

步骤 6 从终点逆向依次搜索其父节点, 直至到达起点, 路径搜索完毕。

基本 RRT 算法伪码如下所示:

基本 RRT 算法主体:

1. $T.init()$;

2. For $k = 1$ to K do;

3. $x_{rand} \leftarrow RANDOM_STATE()$;

4. $x_{nearest} \leftarrow NEAREST_POINT(x_{rand}, T)$

5. $x_{new} \leftarrow NEW_POINT(x_{nearest}, \lambda)$

6. If not $x_{new} \in OBSTACLE$

7. $T.Add_Node()$;

8. end if

9. end for

10. Return T

2 RRT 算法改进

通过对基本 RRT 算法模型和车辆运动学模型的分析, 针对其随机性、路径曲折等问题, 本文提出一种更加高效, 且符合智能车辆运行实际情况的路径规划算法。算法中加入目标偏向策略和密集节点过滤, 如果新节点距离最近节点小于步长 λ , 则认为该节点处于密集区域, 重新生成新节点; 在选择父节点时, 同时考虑距离和车辆转向角约束, 最后对所生成路径进行简化和平滑处理。

2.1 目标偏向策略

基本 RRT 算法是对整个搜索空间进行均匀随机采样, 虽然这种方式能够找到更多的可能路径, 但无针对性, 会导致搜索时间大幅增加, 收敛效率降低, 无法满足智能车的实时性要求。为克服无导向性这一缺陷, 本文采用目标偏向策略, 在采样前生成一随机数 i , 与概率 P (一般不超过 10%) 比较, 满足条件则把目标点 q_{goal} 作为采样点进行扩展, 这种方法不仅保证了算法的随机性, 又能使随机树加速收敛, 向目标点扩展。

$$x_{rand} = \begin{cases} q_{goal}, & i \leq P \\ RANDOM_STATE, & i > P \end{cases} \quad (1)$$

2.2 密集节点过滤

RRT 算法是通过随机采样的方式扩展随机树, 状态空间中的任意位置均有同等概率被节点占据, 随着节点数量增多, 在一定区域内会出现重复扩展, 同一父节点连接多个子节点, 甚至发生重叠现象。大量的重复搜索会增加算法搜索时间, 甚至超过迭代次数, 收敛失败。

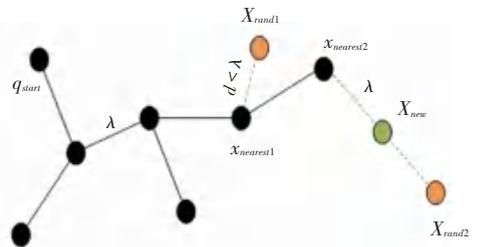


图 2 密集节点过滤示意图

Fig. 2 Schematic diagram of dense node filtering

密集节点过滤, 是对新产生的随机节点做过滤

处理。首先在已扩展节点集 T 中,找到距离新节点 x_{rand} 最近的节点 $x_{nearest}$,在随机树扩展之前判断距离 $dist(x_{rand}, x_{nearest})$ 是否大于步长 λ ,若大于则向 x_{rand} 扩展一定距离,否则舍弃,重新扩展新的节点。如图2所示, x_{rand2} 距离最近节点 $x_{nearest2}$ 大于步长 λ ,正常扩展新节点 x_{new} ;而随机节点 x_{rand1} 到最近节点 $x_{nearest1}$ 的距离 $d < \lambda$,则认为 x_{rand1} 为密集节点,舍弃后重新扩展。

2.3 融合车辆运动学约束的度量函数

在基本 RRT 算法中,其父节点的选择以两点间的欧氏距离作为度量函数,寻找距离新节点最近的节点作为其父节点。但仅仅考虑节点间距离所得到的随机树往往很难满足车辆的行驶要求,因此本文在选择父节点时,充分考虑两节点间的夹角因素,使得到的路径更为平滑。

2.3.1 非完整约束车辆模型

非完整性约束,是指含有系统广义坐标导数且不可积分的约束^[13]。而车辆是一个典型的非完整性约束系统,是在智能车辆自主寻路时必须考虑的因素。空间中,有些区域车辆无法直接到达。在轨迹规划过程中,所得路径必须满足车辆转向特性要求。图3所示为简化车辆运动学模型:

以 $X = (x, y, \theta)$ 表示车辆状态,其中 (x, y) 表示车辆后轮中心的坐标, θ 表示坐标系与车辆前进方向间的夹角。车辆的前后轮轴距为 L , R 为转弯半径,前轮转角为 Φ 且满足 $|\Phi| < \Phi_{max}$ 。由于非完整性约束,车辆做纯滚动运动,且在任何运动瞬间,车辆的速度都必定指向纵轴的垂直方向。假设车辆在平面内做匀速运动,将车辆简化为二自由度自行车模型,可推导得到车辆前轮转角、转弯半径、车辆轴距及曲率之间的关系式。

2.4 步长动态优化

在原始 RRT 算法中,随机树在生成随机点 x_{rand} 后,在父节点以固定步长向随机点扩展。在障碍物密集或搜索边界区域,所产生的新节点发生碰撞或越界的概率较大;而在空旷区域与障碍物碰撞概率大大降低,若仍然采用固定步长,会影响算法计算速度。本文以一种动态步长方法,提高算法的避障能力。即在新节点确定父节点后,首先判断父节点一定范围内是否存在障碍物,若无则判定为安全区域,

$$R = \frac{L}{\tan \varphi} = \frac{1}{k} \quad (2)$$

式中: k 为曲率。

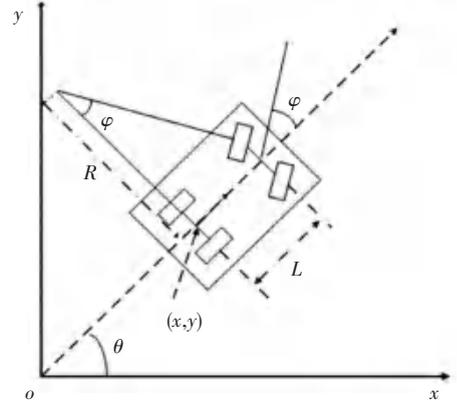


图3 车辆简化运动学模型

Fig. 3 Simplified kinematic model of vehicle

2.3.2 基于车辆最大转向角的父节点选择

如图4所示,随机点选择父节点时既要考虑两点间的距离,同时也要考虑随机点 x_{rand} 与父节点之间的夹角 δ 。当夹角 δ 大于车辆最大转向角 Φ_{max} 时,舍弃新节点 x_{rand} ,重新生成新的节点。夹角 δ 可根据式(3)、式(5)得到。

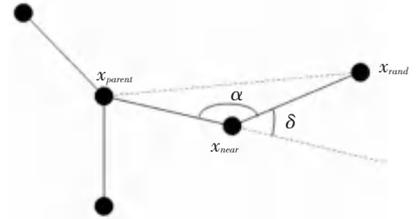


图4 父节点选择示意图

Fig. 4 Parent node selection diagram

$$\delta = \pi - \alpha \quad (3)$$

$$D(i, j) = \sqrt{(i_x - j_x)^2 + (i_y - j_y)^2} \quad (4)$$

$$\alpha = \arccos \frac{D(x_{parent}, x_{near})^2 + D(x_{near}, x_{rand})^2 - D(x_{parent}, x_{rand})^2}{2 \times D(x_{parent}, x_{near}) \times D(x_{near}, x_{rand})} \quad (5)$$

以大步长进行扩展;反之为危险区域,以小步长扩展。新节点位置计算如公式(6)所示。

$$x_{new} = x_{near} + \lambda \times \frac{\Delta P \times (x_{rand} - x_{near})}{\|x_{rand} - x_{near}\|} \quad (6)$$

其中: λ 为固定步长, ΔP 为比例系数,可根据当前父节点位置确定。

2.5 路径的剪枝与平滑

由于 RRT 算法本身的特性,所生成的路径是曲折的,即使在选择父节点时加入转向角约束,仍然存

在很多冗余节点,尤其是在复杂环境下,很难被智能车辆所追踪。本文对已搜索完成所得路径节点进行修剪,并在此采用三次 B 样条曲线对曲线进行曲率优化。

首先,通过改进 RRT 算法找到一条从目标点到起始点的无障碍路径节点集 T ,并创建一个空集合 T_{new} ,将起点加入。然后从初始节点 q_{start} 开始,依次遍历节点集中的所有节点,若两个节点间连线无碰撞,则删除中间的所有多余节点;如果在节点 x_j 发生碰撞,则将节点 x_{j-1} 加入 T_{new} ,然后从 x_{j-1} 个节点开始继续遍历 T ,直到结束并将目标点加入 T 。根据车辆最大转向角约束条件,如果 T_{new} 中相邻路径间夹角 $\alpha < \Phi_{max}$,则将此节点的父节点纳入 T_{new} 中,以满足角度约束。

由于 B 样条曲线具有连续性和局部性等优点,使其在运动规划中的应用广泛^[12]。本文运用三次 B 样条曲线,对上述通过剪枝函数进行处理的路径点进行平滑,生成可被追踪的曲率连续路径。K 阶 B 样条曲线可表示为:

$$P(u) = [P_0 \cdots P_n] \begin{matrix} \hat{e} \\ \vdots \\ \hat{e} \end{matrix} \quad \dot{u} = \sum_{i=0}^n P_i B_{i,k}(u) \quad (7)$$

式中, $B_{i,k}(u)$ 是第 i 个 k 阶 B 样条基函数,与控制点 P_i 相对应; $k \geq 1$; u 是自变量。

2.6 改进 RRT 算法步骤

步骤 1 搜索空间及算法相关参数初始化。

步骤 2 判断是否超过迭代次数,超过寻路失败。

步骤 3 产生一随机数 i ,判断是否满足偏向条件 P ,成立则将目标点作为采样点;反之则产生随机点作为采样点,并判断是否为密集点,满足条件则舍弃。

步骤 4 根据距离最近原则选择父节点,同时判断是否满足车辆最大转向角约束,不满足则舍弃。

步骤 5 根据当前父节点到障碍物距离,动态确定扩展步长。

步骤 6 若新节点无碰撞,则把产生的新节点加入节点集中;并判断该节点是否在目标范围内,成立结束搜索;反之返回步骤 2。

步骤 7 从终点逆向依次搜索其父节点,直至到达起点,路径搜索完毕。

改进 RRT 算法主体如下所示:

改进 RRT 算法主体

1. $T.init()$;

2. For $k = 1$ to K do:

3. if $i < P$:

4. $x_{rand} \leftarrow q_{goal}$

5. else:

6. $x_{rand} \leftarrow RANDOM_STATE()$

7. if not $DENSE_NODE(x_{rand})$

8. $x_{nearest} \leftarrow NEAREST_POINT(x_{rand}, T)$

9. if not $ANGEL_CONSTRAINT(x_{nearest})$

10. $x_{new} \leftarrow NEW_POINT(x_{nearest}, \lambda)$

11. if not $x_{new} \in OBSTACLE$

12. $T.Add_Node()$;

13. end for

14. Return T

3 仿真实验

为验证改进 RRT 算法的优越性和有效性,本文通过 MATLAB 软件进行仿真实验,分别与基本 RRT 算法及单一改进策略 RRT 算法进行对比。

考虑实车运行存在误差等问题,将车辆做膨胀处理。设定车辆为半径 0.3 的圆形,整个搜索空间为 $13 * 12$ 的矩形,车辆初始坐标为 $[0; -1]$,终点坐标是以 $[10.5; 7.5]$ 为中心边长为 1 的正方形,蓝色区域为障碍物,基本 RRT 和目标偏向 RRT 均采用步长 $\lambda = 0.3$ 。

如图 5 所示,图 5(a)~图 5(e)分别为基本 RRT 算法、目标偏向 RRT、节点过滤 RRT、角度约束 RRT、改进 RRT 算法规划结果;图(f)中红色路径为剪枝处理结果;图(g)为三次 B 样条曲线平滑后的路径。

为了正确客观的评价各算法的优劣,利用 MATLAB 构建图 5 中所示的模拟环境,对其分别进行 30 次仿真实验,并记录各算法的运行时间、路径长度和迭代次数,取各指标的平均值进行对比,对比结构见表 1。

表 1 各算法仿真实验对比

Tab. 1 Comparison of simulation experiments of various algorithms

	时间/s	路径长度	迭代次数
RRT	0.573	18.2	306
目标偏向 RRT	0.360	17.5	189
节点过滤 RRT	0.482	18.1	216
角度约束 RRT	0.540	17.9	253
改进 RRT	0.334	16.8	125

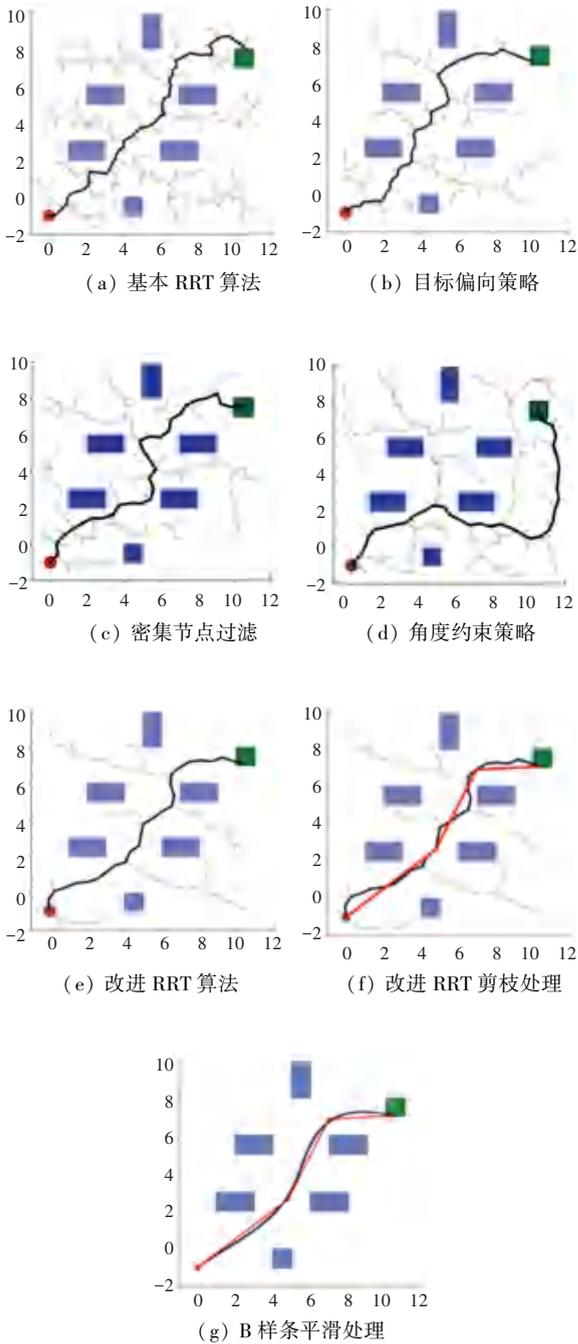


图5 仿真实验结果

Fig. 5 Simulation results

由此可见,本文提出的改进 RRT 算法相较于其它算法花费时间更少,路径长度更短、迭代次数更少。克服了基本 RRT 算法的采样随机性,同时降低了路径的曲折程度,更加符合车辆运动学规律。

4 结束语

本文针对 RRT 算法随机性大、计算效率低、路径曲折等问题,提出一种改进 RRT 算法,首先加入目标点偏向策略并对密集节点进行过滤,加快收敛速度;然后建立车辆运动学模型,增加车辆自身转向约束,并对路径进行优化及平滑,使得规划出的路径更短、曲率更小。最后利用 MATLAB 对 RRT、目标偏向 RRT 和改进 RRT 进行对比仿真测试,验证了改进后算法的可行性。

参考文献

- [1] 余卓平,李奕姗,熊璐. 无人车运动规划算法综述[J]. 同济大学学报(自然科学版),2017,45(8):1150-1159.
- [2] 《中国公路学报》编辑部. 中国汽车工程学术研究报告·2017[J]. 中国公路学报,2017,30(6):1-197.
- [3] 贾庆轩,陈钢,孙汉旭,等. 基于 A* 算法的空间机械臂避障路径规划[J]. 机械工程学报,2010,46(13):109-115.
- [4] 于振中,闫继宏,赵杰,等. 改进人工势场法的移动机器人路径规划[J]. 哈尔滨工业大学学报,2011,43(1):50-55.
- [5] GAMMELL J D, SRINIVASA S S, BARFOOT T D. Informed RRT*: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic[C]//2014 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 2014: 2997-3004.
- [6] LAVALLE S M. Rapidly-exploring random trees: A new tool for path planning[J]. 1998:293-308.
- [7] 施杨洋,杨家富,布升强,等. 基于 RRT 改进的智能车辆路径规划算法[J]. 计算技术与自动化,2019,38(4):81-86.
- [8] DU M, CHEN J, ZHAO P, et al. An improved RRT-based motion planner for autonomous vehicle in cluttered environments[C]//2014 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2014: 4674-4679.
- [9] KARAMAN S, FRAZZOLI E. Sampling-based algorithms for optimal motion planning[J]. The international journal of robotics research, 2011, 30(7): 846-894.
- [10] 刘成菊,韩俊强,安康. 基于改进 RRT 算法的 RoboCup 机器人动态路径规划[J]. 机器人,2017,39(1):8-15.
- [11] QURESHI A H, MUMTAZ S, IQBAL K F, et al. Triangular geometry based optimal motion planning using RRT* - motion planner[C]//2014 IEEE 13th International Workshop on Advanced Motion Control (AMC). IEEE, 2014: 380-385.
- [12] 李红,王文军,李志强. 基于 B 样条理论的平行泊车路径规划[J]. 中国公路学报,2016,29(9):143-151.
- [13] 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.