

# 复杂环境下基于目标指引的 RRT \* 路径规划算法<sup>①</sup>

魏立新<sup>②\*\*\*</sup> 侯仕杰<sup>\* \*\*\*</sup> 孙 浩<sup>\* \*\*\*</sup> 吴绍坤<sup>\*\*\*</sup>

(<sup>\*</sup> 燕山大学智能控制系统与智能装备教育部工程研究中心 秦皇岛 066004)

(<sup>\*\*</sup> 燕山大学工业计算机控制工程河北省重点实验室 秦皇岛 066004)

(<sup>\*\*\*</sup> 天津电气科学研究院有限公司 天津 300180)

**摘要** 机器人路径规划算法需应对运动过程中遇到的各种复杂环境。针对快速扩展随机树 (RRT) 算法规划时间长、产生新节点随机性大、盲目性强的缺点, 提出基于目标指引的 RRT \* 路径规划算法。该算法在障碍物和目标点处分别产生虚拟势场, 引入引力函数和斥力函数使得生成的随机点具有目标性, 随机点朝向目标点方向产生, 降低盲目性和随机性; 回归策略和动态自适应步长策略减少规划时间和产生冗余点的数量。当环境复杂时, 提出带有预测机制的模糊推理策略, 以解决机器人在 U 型陷阱下易产生的局部死锁现象。在动态环境下, 提出重规划策略使机器人拥有动态避障能力。最后, 在树莓派智能小车上进行了实验测试, 结果验证了该算法的有效性。

**关键词** 复杂环境; 目标指引 RRT \* ; 模糊推理策略; 动态环境; 树莓派智能小车

## 0 引言

正在实施的中国制造 2025 战略提倡发展机器人技术。机器人路径规划是智能移动机器人的核心技术之一<sup>[1]</sup>, 从 20 世纪 70 年代开始, 已有大量移动机器人路径规划技术运用到生活中<sup>[2]</sup>。作为路径导航最主要的部分, 移动机器人路径规划<sup>[3]</sup>技术的发展速度日益提升。其核心内容为在有障碍物的环境中, 规划出一条从起始点到目标点无碰撞的路径<sup>[4]</sup>。

机器人路径规划依据地图环境可分为全局路径规划和局部路径规划<sup>[5]</sup>。常用的全局路径规划算法有 A \*<sup>[6]</sup> 算法、D \*<sup>[7]</sup> 算法、快速扩展随机树 (rapidly-exploring random trees, RRT) 算法<sup>[8]</sup>、拓扑法<sup>[9]</sup> 和 RRT \* 算法<sup>[10]</sup> 等。根据环境中障碍物的形态可分为动态环境<sup>[11]</sup> 和静态环境<sup>[12]</sup>。其中 A \* 和 D \* 算

法利用启发式思想能够分别在动态环境和静态环境中规划出一条最优路径, 但其计算量大、规划时间长、效率低。RRT 算法<sup>[13]</sup> 由于其特殊的采样性搜索进入人们的视野, 它的优点在于搜索速度快、效率高和搜索范围广, 缺点是导致路径不是渐近最优、随机性大和冗余点多。因此文献[14]在基于 RRT 算法的基础上提出了 RRT \* 算法, 该算法在 RRT 基础上增加了父节点的重新选择, 改进后每一阶段都达到渐进最优的路径。由于每个节点都需父节点的重新选取, 导致计算量增加、规划时间增长并且效率降低。针对上述不足, 国内外学者对其进行了不同程度上的改进。文献[15]提出了一种基于高斯混合模型的高维构型空间碰撞检测算法, 应用于 RRT \* 算法中, 使用增量期望最大化算法学习基于环境的高斯混合模型, 用于代替传统方法进行碰撞检测, 有效地减少了碰撞检测的次数, 提高了 RRT \* 算法的效率。但收敛性没有得到改进。文献[16]提出了

① 国家重点研发计划(2018YFB1702300)资助项目。

② 男, 1977 年生, 博士, 教授; 研究方向: 机器人控制技术, 冶金自动化, 智能数控; 联系人, E-mail: wlx2000@ysu.edu.cn  
(收稿日期: 2020-07-06)

Q-RRT \* 算法,该算法收敛速度快,并且通过扩大父节点集降低计算成本,加快接近目标点。但在复杂环境下难以完成规划,当机器人具有运动学约束并且成本不满足三角不等式时算法受限。文献[17]对于高斯混合模型(Gaussian mixture model, GMM)的 RRT \* 算法仅适用于单一固定场景的局限性,引入记忆机制,提出了一种融合长短时记忆机制的快速运动规划算法用来解决多场景下的快速运动规划问题。文献[18]提出了在动态环境下规划路径算法,减少了原 RRT 算法的路径长度,但规划的速度较慢、随机性大。文献[19]提出了连续曲率 RRT 算法,提升了规划的速度和质量,解决了在复杂环境下路径规划问题,但规划的路径并不是最优或渐近最优。针对未知环境下移动机器人路径规划,文献[20]提出了基于子目标搜索的目标导向 RRT 算法,通过设置目标函数减小盲目性,提升了规划效率,子目标搜索策略避免了产生极小值<sup>[21]</sup>和未知环境规划问题<sup>[22]</sup>,但仍产生较多的冗余点,且未考虑到动态环境的情况。

本文针对 RRT \* 算法在静态环境下规划时间缓慢和盲目性大的问题,提出了一种基于目标指引的 RRT \* 路径规划算法。该算法以新节点产生位置为改进重点,结合引力函数和斥力函数使新节点朝目标点方向产生。引入回归策略和自适应步长策略筛选新节点位置,回归不符合要求的节点来减少“树枝”冗余度和路径长度。复杂环境下,结合带有预测机制的模糊推理策略解决局部死锁问题。最后,结合重规划策略以解决动态环境<sup>[23]</sup>下路径规划问题。

## 1 改进 RRT \* 算法

### 1.1 基本 RRT \* 算法

RRT \* 是基于 RRT 算法的改进,无需对全局环境进行准确建模。其核心内容以产生的随机点为圆心,在以  $r$  为半径的圆范围内,选取备选父节点,计算其随机点到备选父节点再到起始点的代价,并对比随机点到原父节点到起始点的代价。如果前者的代价小于后者,则更新随机点的父节点,反之不更

新。依次对“树”上的所有节点进行上述步骤。全部更新完成后,新的路径比原 RRT 算法规划出的路径长度明显减短,路径渐近最优。如图 1(a) 和图 1(b) 所示,  $X_{init}$  为起始点,  $X_{rand}$  为随机点,  $X_{parent}$  为原父节点,  $X_2$  和  $X_3$  为备选父节点, 黑色矩形为障碍物。经过筛选  $X_3$  为新父节点。但由于 RRT \* 算法在规划路径时需重新选择父节点,导致规划时间与计算量增加,且随机性大。本文依据上述不足进行了改进。

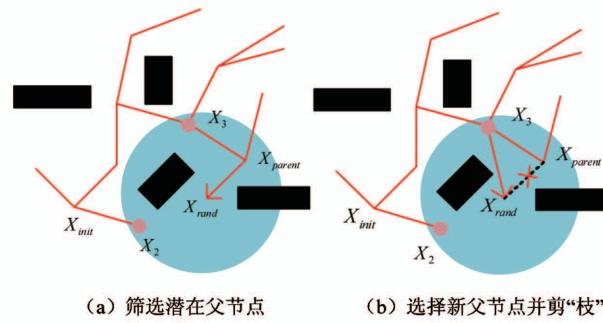


图 1 重新选取父节点

### 1.2 目标指引 RRT \* 算法

#### 1.2.1 改进后新节点的选取方式

针对 RRT \* 算法产生新节点的随机性大、盲目性强、规划时间长的缺点,本文采用目标指引 RRT \* 路径规划算法。该算法利用模拟人工势场法虚拟势场的思想,引入了引力函数和斥力函数。

引力函数如式(1)所示。

$$U_{att} = \frac{1}{2} \lambda (X - X_0)^2 \quad (1)$$

式中,  $\lambda$  为引力势场正比例系数,  $(X - X_0)^2$  表示当前点  $X$  到目标点  $X_0$  的距离长度。 $X$  点处的引力为该点势场的负梯度, 引力如式(2)所示。

$$F_{att} = -\text{grad}\left(\frac{1}{2} \lambda (X - X_0)^2\right) = \lambda (X - X_0) \quad (2)$$

机器人在斥力的影响下,使其远离障碍物所在的位置,当机器人与障碍物超过排斥力势场影响的范围后,其斥力为 0 不影响机器人运动。斥力函数如式(3)所示。

$$U_{rep} = \begin{cases} \frac{1}{2} \eta \left( \frac{1}{\rho} - \frac{1}{\rho_0} \right) & \rho \leq \rho_0 \\ 0 & \rho > \rho_0 \end{cases} \quad (3)$$

式中,  $\eta$  为斥力势场正比例系数;  $\rho$  代表当前位置与

障碍物之间的距离长度;  $\rho_0$  为障碍物影响范围, 当机器人与障碍物距离大于  $\rho_0$ , 斥力势场不影响障碍物。产生的斥力为斥力势场负梯度, 斥力函数表达式如式(4)所示。

$$\begin{aligned} F_{rep} &= -\text{grad}(U_{rep}) \\ &= \begin{cases} \left(\frac{1}{\rho} - \frac{1}{\rho_0}\right)\eta \frac{1}{\rho^2} \frac{\partial \rho}{\partial X} & \rho \leq \rho_0 \\ 0 & \rho > \rho_0 \end{cases} \quad (4) \end{aligned}$$

式中,  $\frac{\partial \rho}{\partial X}$  表示障碍物位置指向  $X$  当前点位置的单位向量, 在复杂环境下, 如式(5)所示。

$$\theta_n = \arctan\left(\frac{y_n - y}{x_n - x}\right) \quad (5)$$

其中,  $\theta_n$  表示第  $n$  个障碍物对机器人产生斥力的方向角度,  $(x_n, y_n)$  为第  $n$  个障碍物坐标,  $(x, y)$  为机器人当前坐标。下式为目标点所在方向角  $\theta_{goal}$ :

$$\theta_{goal} = \arctan\left(\frac{y_{goal} - y}{x_{goal} - x}\right) \quad (6)$$

式(7)和式(8)分别为经过受力分析在  $x$  轴和  $y$  轴的分力。

$$F_x = F_{att} \cos \theta_{goal} - (F_{rep1} \cos \theta_1 + \cdots + F_{repn} \cos \theta_n) \quad (7)$$

$$F_y = F_{att} \sin \theta_{goal} - (F_{rep1} \cos \theta_1 + \cdots + F_{repn} \cos \theta_n) \quad (8)$$

经数学公式得出下式新节点产生的方向角度:

$$\theta = \arctan\left(\frac{F_y}{F_x}\right) \quad (9)$$

改进后 RRT \* 产生的新节点公式为

$$X_{newx} = X_x + l \times \cos \theta \quad (10)$$

$$X_{newy} = X_y + l \times \sin \theta \quad (11)$$

式(10)和式(11)中  $l$  为步长,  $X_{new}$  为新节点坐标,  $X$  为当前点坐标, 与原算法相比改进后算法的新节点产生由引力常数  $\lambda$ 、斥力常数  $\eta$  和步长  $l$  共同决定( $l$  由式(13)决定)。

新节点产生图示如图 2 所示。

图 2 中圆点为 RRT \* “树”上的节点,  $X_{init}$  为起始点,  $X_{new}$  为产生的新节点,  $G$  为目标点, 箭头所指方向为产生力的方向, 其中  $F_{all}$  方向延伸虚线为随机点产生的方向, 根据设定的步长, 确定新节点产生的位置。

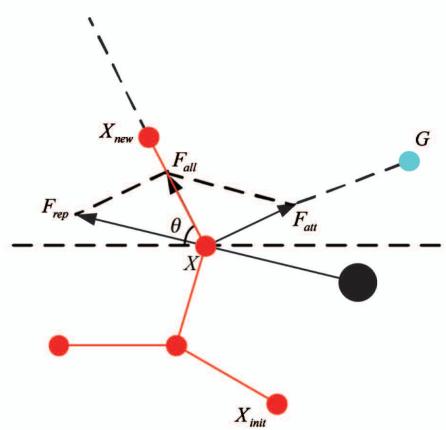


图 2 目标指引 RRT \* 新节点生成

### 1.2.2 回归策略

由于空间的连续性, 明确计算已探索的配置空间是不现实的, 很难完全描述随机树所覆盖的配置空间并实现稳健的回归约束。RRT \* 算法在规划中产生较多的冗余点, 大量空白区域和已搜索区域搜索重复搜索。本文针对上述不足提出回归策略, 该策略通过调整半径来改变回归的密集程度。当新配置  $x_{new}$  与随机树上最近节点的距离小于  $x_{new}$  与父  $x_{parent}$  之间的距离时, 则配置  $x_{new}$  被认为是回归的。进一步说明了随机树扩展过程的机制, 式(12)为提取需要回归点的规则。

$$\begin{cases} \exists x \in \text{Tree} \\ \rho(x, x_{new}) < k\varepsilon(x_{parent}, x_{new}) \end{cases} \quad (12)$$

式中,  $(x, x_{new})$  表示  $x_{new}$  与“树”上其他节点  $x$  的代价,  $(x_{parent}, x_{new})$  表示  $x_{new}$  与  $x_{parent}$  之间的代价,  $\varepsilon$  为步长。如图 3 所示, 空心圆节点  $x_3, x_4$  将回归, 只有最接近其父节点的  $x_1$  或  $x_2$  节点才能正常扩展。对于产生回归节点, 本文采用目标偏向采样策略重新选择新节点。该策略根据设定的目标偏向阈值  $\gamma_{goal}$

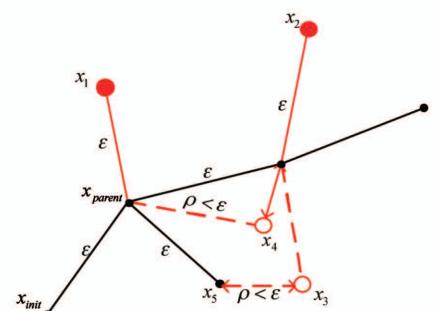


图 3 回归策略

与均匀概率分布的概率值  $\gamma$  进行对比。如果前者大于后者, 则  $\gamma_{goal} = \gamma$ , 否则会生成一个随机节点  $x_{rand}$ 。该机制可以有效地防止过度搜索已经探索过的区域, 减少产生不必要的冗余点。

### 1.2.3 自适应步长策略

由于 RRT \* 算法计算量大, 使得规划时间增长, 并且规划的路径不是平滑曲线。本文在基于上述小节得知新节点产生的方向, 加入动态自适应步长策略使得当前点与新节点之间无障碍物的情况下, 直接扩展两点之间作为当前步长, 缩短了固定步长而增加的路径长度。使得路径平滑且路径总转向角度减少。如图 4 所示, 以  $X_{nearest}$  为原点和  $X_{new}$ 、 $X_{goal}$  分别形成两个向量, 并且计算出两向量之间的夹角  $\alpha$ 。动态自适应步长的长度计算公式如式(13)所示。

$$\begin{cases} \rho = \rho_{min} + (\rho_{max} - \rho_{min}) \times \cos\alpha & \alpha \leq 90^\circ \\ \rho = \rho_{min} & \alpha > 90^\circ \end{cases} \quad (13)$$

其中  $\rho_{min}$  和  $\rho_{max}$  分别为最短扩展步长  $l = 1$  和机器人受影响范围内目标方向的障碍物之间最长扩展步长。图 4 为式(13)的图示。

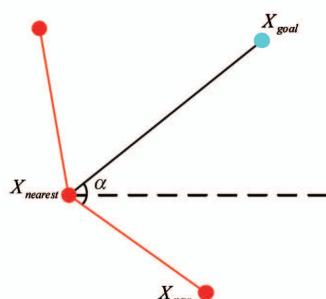


图 4 动态自适应步长选取策略

### 1.2.4 动态环境下重规划策略

移动机器人不仅在静态环境下规划路径, 也要在动态环境下进行路径规划。现实中行走的人、移动的汽车等都可看作动态障碍物, 在机器人行进过程中要避开。本文针对动态环境下机器人路径规划提出了基于目标指引的 RRT \* 算法的重规划策略, 该策略主要依据二次规划的想法, 由于 RRT \* 算法先规划路径, 再指导机器人沿路径行走。当移动障碍物移动到已规划完的路径上时, 机器人探测到路

径被挡后, 立即以该点作为起始点, 在已规划好的路径上进行重新规划, 绕过障碍物后重新连接原路径。该策略过程如图 5 所示, 虚线为已规划并且被障碍物挡住的路径, 实线为重规划路径, 圆形为障碍物。

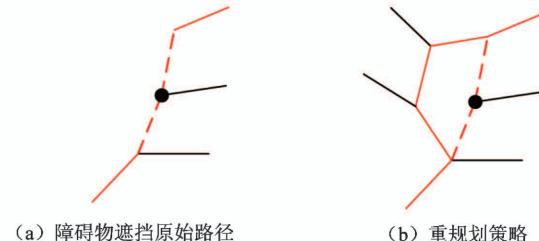


图 5 重规划策略

## 2 复杂环境下解决目标不可达现象

### 2.1 问题描述

由于环境是复杂多变的, 障碍物的位置信息形态存在不确定性, 以地图环境中是否存在移动障碍物可分为动态环境和静态环境。由于存在复杂 U 型障碍物易使得小车陷入局部死锁现象, 无法到达目标点, 因此提出带有预测机制的模糊推理策略进行逃离。

### 2.2 设置模糊控制器

运用通用模糊控制依据传统方式进行设计。模糊控制中进行模糊推理的是模糊变量, 而机器人所处环境中各方向的障碍物距离信息、目标方向信息都是具体的量, 因此存在一个转换的过程。采用简单线性化处理的方法, 将障碍物距离输入变量定为  $DL, DC, DR$ , 其模糊语言为  $\{ND, MD, FD\}$ , 论域为  $0 \sim 7$ ; 机器人当前运动方向和目标点之间夹角为  $TDA$ , 其模糊语言为  $\{L, M, R\}$ , 论域为  $-180^\circ \sim 180^\circ$ 。输出为随机点产生的区域  $Place$ , 模糊语言及论域与目标方位角相同。以上各变量隶属度函数均为三角形。基于模糊控制器的模糊控制库共有 81 条控制规则, 由于篇幅原因, 此处列举了其中 5 条规则, 如表 1 所示。并采用成熟的 Mamdani 模糊控制方法, 运用重心法进行解模糊。

### 2.3 带有预测机制的模糊推理策略

为解决环境中存在 U 型陷阱障碍物而产生局部死锁现象, 本文在已改进的算法基础上结合带有预测机制的模糊推理策略。该策略随预测条件的改

表 1 模糊推理规则库

序号	输入				输出
	DL	DC	DR	TDA	
1	ND	ND	ND	L	L
2	ND	ND	MD	R	R
3	FD	MD	MD	R	R
4	FD	MD	FD	R	R
5	FD	FD	FD	M	M

变,其输入目标方位角改变,从而使得新节点的选取位置发生改变。具体预测条件如下,当处于 U 型陷阱中的机器人受到合外力为 0 或连续 3 个目标方位角大于正 90°或小于负 90°,也就是转向角绝对值

总和大于 180°时,根据式(14)重新计算新的目标方位角,使之逃出 U 型陷阱,式(14)中 angle 为目标方位角(TDA)。

$$angle = \begin{cases} angle + \pi & angle \in (0, \frac{\pi}{2}) \\ angle - \pi & angle \in [-\frac{1}{2}, 0] \end{cases} \quad (14)$$

该策略使机器人在行进过程中产生局部死锁现象时,能够顺利逃出。

改进算法总流程图如图 6 所示。

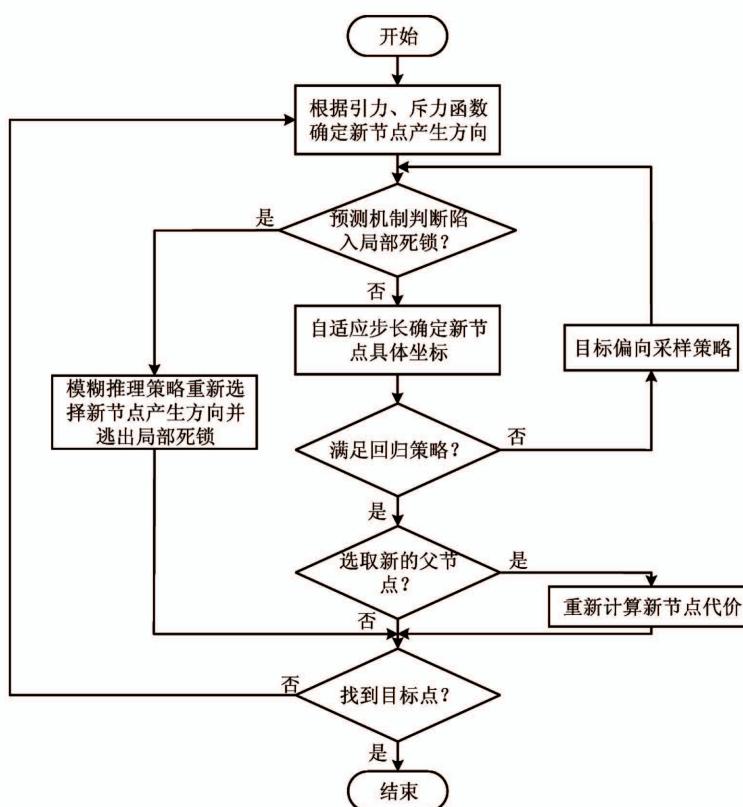


图 6 改进算法总流程图

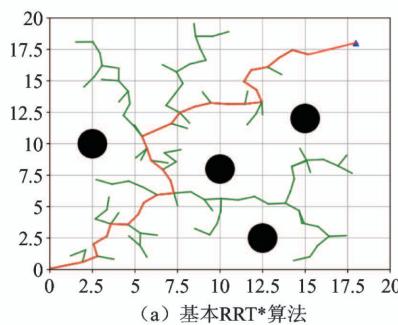
### 3 仿真与实验

为验证本文算法的正确性和有效性,利用 Python 语言在 Pycharm 平台上进行仿真,并在树莓派智能小车上进行实验验证。在仿真中简化智能小车为小圆点,仿真参数如下,与障碍物最小距离 D 为 0.03,地图大小均采用 20×20 的动态环境和静态环境。

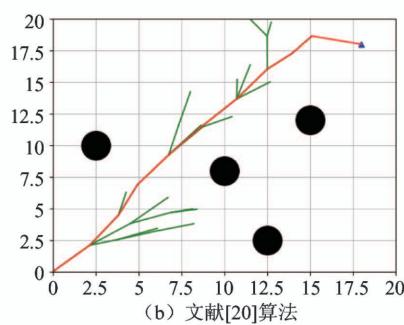
#### 3.1 目标指引 RRT \*

##### 3.1.1 静态环境

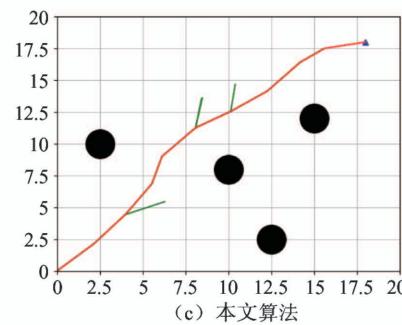
在静态环境下,起始点坐标设为(0, 0),目标点坐标设为(18, 18)。大小为 2 的圆形障碍物分布在正方形边长为 20 的地图中。将本文算法(图 7(c))与基本 RRT \* 算法(图 7(a))、文献[20]目标导向 RRT 算法(图 7(b))仿真结果做对比。可以看出本文算法方向性增强、冗余点减少、路径长度减短。



(a) 基本RRT\*算法



(b) 文献[20]算法



(c) 本文算法

图 7 算法对比图

另从表 2 中所示 50 组已知静态环境下仿真实验数据,均取平均值。可看出本文算法的搜索时间、搜索路径规划长度和“树”的分支数均较基本 RRT \* 算法和文献[20]算法明显改进。

表 2 算法性能对比

算法	时间/s	分支数	长度/cm
基本 RRT * 算法	1.40	105	32
文献[20]算法	0.49	47	29
本文算法	0.43	30	27

### 3.1.2 实物实验

为更好地验证本文算法的优越性,如图 8 所示,在搭载 RaspberryPi3B + 轮式四驱差速智能小车上进行实验。

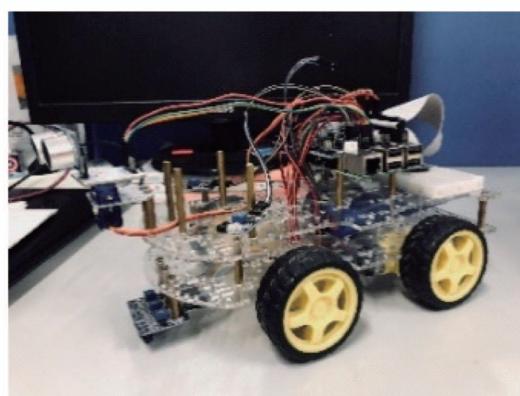


图 8 树莓派智能小车

如图 9 所示,仿真参数如下,起始点坐标为(0, 0), 目标点坐标为(15, 9)。其中水瓶均为直径 6 cm 的障碍物。

图9(a)和图9(b)分别为文献[20]算法和本文

算法在图 9(c)和图 9(d)路径下实验小车运行过程某位姿。图 9(c)为文献[20]路径仿真图,图 9(d)为改进算法路径仿真图。在实物实验中,其地图为边长为 200 cm,在实验中忽略树莓派小车体积。

表 3 为树莓派小车 15 次实验结果平均计算得出。从表中可看出改进算法在小车运行时间和运行总长度均优于 RRT \* 算法和文献[20]算法,进一步证明了本文算法的优越性。

表 3 树莓派小车实验对比数据

算法	小车运动时间/s	小车运行路径长度/cm
基本 RRT * 算法	41.0	215.77
文献[20]算法	29.6	196.15
本文算法	23.5	186.32

在实物实验中,由于本文算法未考虑到树莓派小车体积,易与路径距离较近的障碍物发生碰撞。另外,没有考虑到障碍物的形状,本文算法对于存在不规则障碍物环境下仍有局限性。

### 3.2 带有预测机制的模糊推理策略仿真研究

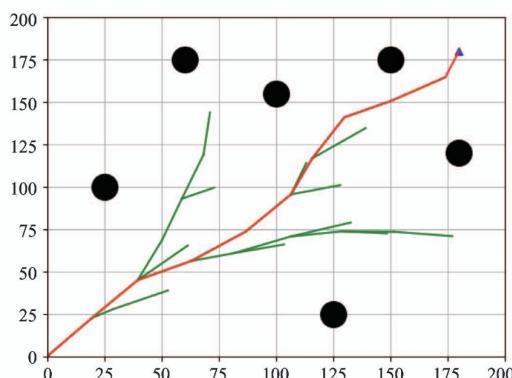
如图 10 所示,该 U 型陷阱环境中,机器人起始点  $S = (10, 10)$ , 目标点  $G = (18, 18)$ 。图 10(a)为产生第 1 个目标方位角大于正  $90^\circ$  或小于  $-90^\circ$  的随机节点,图 10(b)、10(c)分别为连续产生第 2 个和第 3 个目标方位角大于正  $90^\circ$  或小于负  $90^\circ$  的随机节点,且在图 10(c)中陷入局部死锁。累计转角角度绝对值大于  $180^\circ$ ,导致机器人无法逃离陷阱,满足预测机制条件。图 10(d)为启动该机制,使得机器人快速逃离 U 型陷阱,完成规划。



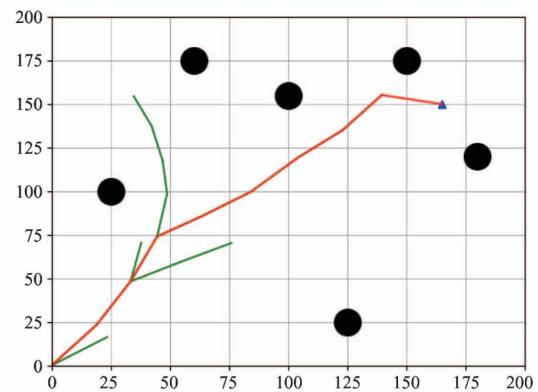
(a) 文献[20]算法实验某位姿



(b) 本文算法实验某位姿

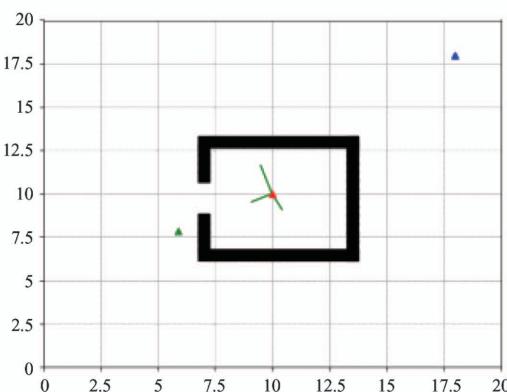


(c) 文献[20]算法仿真图

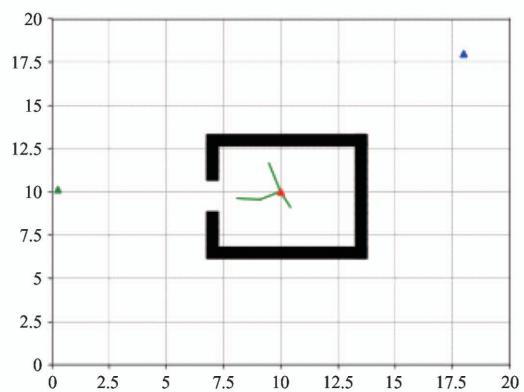


(d) 本文算法仿真图

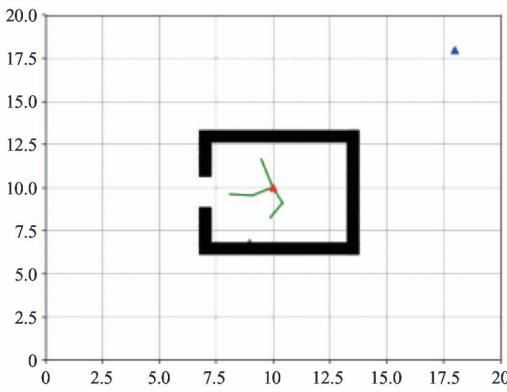
图9 实验中的小车及仿真图



(a) 目标方位角大于 90°



(b) 目标方位角连续大于 90°



(c) 陷入局部死锁



(d) 逃出局部死锁

图10 模糊推理策略逃出局部死锁

## 4 结 论

提出了一种复杂环境下基于目标指引的 RRT \* 路径规划算法。引入引力斥力函数、回归策略和自适应步长策略使新节点具有方向性,减少冗余点的产生,降低了计算量和规划时间。在此基础上,结合带有预测机制的模糊推理策略,以应对 U 型陷阱的局部死锁情况。在动态环境下,结合重规划策略躲避移动障碍物。将本文算法与现有算法进行对比,验证了其优越性和有效性。

本文算法主要针对在已知环境下规划路径,无法做到实时规划,未来将研究重点放于结合传感器及摄像头进行实时规划路径。

## 参 考 文 献

- [ 1 ] Ulises O R, Oscar M, Roberto S. Mobile robot path planning using membrane evolutionary artificial potential field [ J ]. *Applied Soft Computing Journal*, 2019, 77: 236-251
- [ 2 ] Zhang Y Y, Li S. Distributed biased min-consensus with applications to shortest path planning [ J ]. *IEEE Transactions on Automatic Control*, 2017, 62( 10 ): 5429-5436
- [ 3 ] 余文凯,章政,付雪画,等. 基于地图预处理及改进 A \* 算法的路径规划 [ J ]. 高技术通讯, 2020, 30( 4 ): 383-390
- [ 4 ] Raj A, Thakur A. Dynamically feasible trajectory planning for Anguilliform-inspired robots in the presence of steady ambient flow [ J ]. *Robotics and Autonomous Systems*, 2019, 118: 144-158
- [ 5 ] 朱大奇,颜明重. 移动机器人路径规划技术综述 [ J ]. 控制与决策, 2010, 25( 7 ): 961-967
- [ 6 ] 赵晓,王铮,黄程侃,等. 基于改进 A \* 算法的移动机器人路径规划 [ J ]. 机器人, 2018, 40( 6 ): 903-910
- [ 7 ] 史久根,刘春霞,席海强. CA 模型下的改进 D \* 路径规划算法 [ J ]. 电子测量与仪器学报, 2016, 30( 1 ): 30-37
- [ 8 ] Kuffner J J, LaValle S M. RRT-connect: an efficient approach to single-query path planning [ C ] // IEEE International Conference on Robotics and Automation ( ICRA ), San Francisco, USA, 2000: 1-7
- [ 9 ] 闫飞,庄严,白明,等. 基于拓扑高程模型的室外三维环境建模与路径规划 [ J ]. 自动化学报, 2010, 36( 11 ): 1493-1501
- [ 10 ] 朱宏辉,明瑞冬,朱轶. 基于改进 RRT \* 算法的路径规划 [ J ]. 武汉理工大学学报, 2017, 39( 2 ): 72-76
- [ 11 ] 江超,邢科新,林叶贵,等. 未知环境下移动机器人静态与动态实时避障方法研究 [ J ]. 高技术通讯, 2019, 29( 10 ): 1012-1020
- [ 12 ] Malone N, Chiang H T, Lesser K, et al. Hybrid dynamic moving obstacle avoidance using a stochastic reachable set-based potential field [ J ]. *IEEE Transactions on Robotics*, 2017, 33( 5 ): 1124-1138
- [ 13 ] Zhang H J, Wang Y K, Zhang J, et al. Path planning of industrial robot based on improved RRT algorithm in complex environments [ J ]. *IEEE Access*, 2018, 6: 53296-53306
- [ 14 ] Karaman S, Frazzoli E. Sampling-based algorithms for optimal motion planning [ J ]. *International Journal of Robotics Research*, 2011, 30( 7 ): 846-894
- [ 15 ] Jinwook H, Lee D D. Learning high dimensional mixture models for fast collision detection in rapidly exploring random trees [ C ] // IEEE International Conference on Robotics and Automation, Stockholm, Sweden, 2016: 62-69
- [ 16 ] Jeonga I B, Lee S J, Kim J H. Quick-RRT \* : triangular inequality-based implementation of RRT \* with improved initial solution and convergence rate [ J ]. *Expert Systems with Applications*, 2019, 123: 82-90
- [ 17 ] 张云洲,孙永生,夏崇坤,等. 融合长短时记忆机制的机械臂多场景快速运动规划 [ J ]. 控制与决策, 2020, 35( 12 ): 2968-2976
- [ 18 ] 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
- [ 19 ] 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 ): 443-449
- [ 20 ] 阮小钢,周静,张晶晶,等. 基于子目标搜索的机器人目标导向 RRT 路径规划算法 [ J ]. 控制与决策, 2020, 35( 10 ): 2543-2548
- [ 21 ] 魏立新,吴邵坤,孙浩,等. 基于多行为的移动机器人路径规划 [ J ]. 控制与决策, 2019, 34( 12 ): 2721-2726
- [ 22 ] 朱毅,张涛,宋婧雁. 未知环境下势场法路径规划的局部极小问题研究 [ J ]. 自动化学报, 2010, 36( 8 ): 1122-1130
- [ 23 ] Drake D, Koziol S, Chabot E. Mobile robot path planning with a moving goal [ J ]. *IEEE Access*, 2018, 6: 12800-128146

# RRT \* path planning algorithm based on target guidance in complex environment

Wei Lixin<sup>\* \*\*\*</sup>, Hou Shijie<sup>\* \*\*\*</sup>, Sun Hao<sup>\* \*\*\*</sup>, Wu Shaokun<sup>\*\*\*</sup>

(<sup>\*</sup> Engineering Research Center for Intelligent Control System and Intelligent Equipment,  
Ministry of Education, Yanshan University, Qinhuangdao 066004)

(<sup>\*\*</sup> Hebei Key Laboratory of Industrial Computer Control Engineering, Yanshan University, Qinhuangdao 066004)

(<sup>\*\*\*</sup> Tianjin Electric Science Research Institute Co. Ltd, Tianjin 300180)

## Abstract

Robot path planning algorithm needs to deal with various complex environments encountered in the process of movement. According to the disadvantages of the algorithm of rapidly-exploring random trees(RRT), such as long planning time, large randomness and strong blindness of new nodes, an algorithm of RRT \* path planning based on target guidance is proposed. In this algorithm, virtual potential fields are generated at obstacles and target points respectively. By introducing the gravitational function and the repulsive function, the generated random points are targetable. The random points are generated towards the target point, which reduces the blindness and randomness. Regression and dynamic adaptive step-size strategies reduce planning time and the number of redundant points. When the environment is complex, a fuzzy inference strategy with predictive mechanism is proposed to solve the local deadlock in U-shaped traps. In dynamic environment, the reprogramming strategy is proposed to enable the robot to have dynamic obstacle avoidance ability. Finally, the experiment is carried out on the Raspberry Pi smart car. The results show that the algorithm is effective.

**Key words:** complex environment, target guidance RRT \*, fuzzy inference strategy, dynamic environment, Raspberry Pi smart car